

CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO



Sistema de identificação de direção perigosa utilizando inteligência artificial e MPU-6050

Aluno: Luisa Nogueira Costa Cavalcante(lncc@cin.ufpe.br)

Orientador: Adriano Augusto de Moraes Sarmento (aams@cin.ufpe.br)

Área: Sistemas Embarcados, Internet das coisas e ciência de dados

Recife, 2023

UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE INFORMÁTICA

LUISA NOGUEIRA COSTA CAVALCANTE

SISTEMA DE IDENTIFICAÇÃO DE DIREÇÃO PERIGOSA UTILIZANDO ACELERAÇÃO E VELOCIDADE
ANGULAR COLETADOS PELO MPU-6050

TRABALHO APRESENTADO AO PROGRAMA DE CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO DO CENTRO DE INFORMÁTICA DA UNIVERSIDADE FEDERAL DE PERNAMBUCO COMO REQUISITO PARCIAL PARA OBTENÇÃO DO GRAU DE BACHAREL EM ENGENHARIA DA COMPUTAÇÃO.

ORIENTADOR: PROF Adriano Augusto de Moraes Sarmiento

Recife

18 de setembro de 2023

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Cavalcante, Luisa Nogueira Costa.

Sistema de identificação de direção perigosa utilizando inteligência artificial e MPU-6050 / Luisa Nogueira Costa Cavalcante. - Recife, 2023.

38 p. : il., tab.

Orientador(a): Adriano Augusto de Moraes Sarmento

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Informática, Engenharia da Computação - Bacharelado, 2023.

9.5.

1. Sistemas Embarcados. 2. Inteligência Artificial. 3. Internet das coisas. 4. Ciência de dados. I. Sarmento, Adriano Augusto de Moraes. (Orientação). II. Título.

000 CDD (22.ed.)

Agradecimentos

Gostaria de expressar minha profunda gratidão a todas as pessoas e instituições que contribuíram para a realização deste trabalho de conclusão de curso. Este projeto representou o fechamento da minha graduação, e não teria sido possível sem o apoio, orientação e incentivo que recebi ao longo desta jornada acadêmica.

Primeiramente, quero agradecer ao meu orientador, Professor Adriano Sarmiento, pela expertise, orientação e acompanhamento que proporcionou durante todo o processo de pesquisa. Suas valiosas sugestões e conselhos foram fundamentais para aprimorar este trabalho.

Aos colegas Igor Moura, João Gabriel Vasconcelos e Maria Eduarda Barros pela amizade e por me ajudarem diretamente em dúvidas no desenvolvimento dos algoritmos de Inteligência Artificial deste projeto.

À professora Edna Barros, que abriu portas para que eu, junto com minha equipe, pudesse participar de competições de Sistemas Embarcados e me aprofundar na área, além de fornecer todo o apoio necessário durante todo o desenvolvimento dos projetos.

Aos meus amigos da Estufa por estarem sempre à disposição para ajudar em qualquer ideia que surgisse.

Aos meus companheiros de jornada do Herói Cuidador e da Hábitus, por aceitarem um desafio que muitas vezes parecia ser inalcançável e construirmos juntos uma solução inovadora.

Às amigadas que cultivei ao longo desta jornada acadêmica, obrigado por compartilharem os altos e baixos dessa jornada comigo.

Ao meu pai, Valber, por ser meu exemplo de determinação e criatividade. Como um engenheiro nato, sempre apoiando minhas novas ideias ou projetos. À minha mãe, Daniela, por sempre me apoiar, nunca me fazer desistir e por me incentivar a dar o meu melhor todos os dias. Cada conquista que alcancei é em grande parte resultado do amor e apoio que recebi de vocês.

Aos meus amigos, obrigado por todas as histórias; sua amizade e trabalhos juntos tornaram esta jornada mais significativa e memorável.

Por fim, gostaria de agradecer ao Centro de Informática da UFPE pela ótima formação acadêmica que recebi.

Este TCC representa não apenas meu esforço individual, mas também a contribuição de muitas pessoas e instituições que acreditaram em mim e me apoiaram ao longo desta jornada. A todos vocês, meu mais profundo agradecimento.

Resumo

Esse trabalho apresenta um sistema IoT para detecção de direção perigosa utilizando técnicas de inteligência artificial, sistema embarcado e um painel de gestão para visualização dos resultados. O sistema embarcado é descrito juntamente com os algoritmos de inteligência artificial avaliados, as bases de dados avaliadas e os dados coletados pelo sistema embarcado. O sistema utiliza tanto uma base de dados coletada pelo telefone como dados coletados por sensores no sistema embarcado. Foi realizado experimentos com cinco algoritmos de inteligência artificial. O trabalho é inovador por trazer uma análise temporal da condução do motorista.

Palavras-chave: Sistemas Embarcados, Inteligência artificial, Direção perigosa, IoT

Abstract

This work presents a system for detecting dangerous driving using artificial intelligence techniques, an embedded system and a management panel to visualize the results. The embedded system is described along with the evaluated artificial intelligence algorithms, the evaluated databases and the dataset collected by the embedded system. The system uses both a database collected by the phone and data collected by sensors in the embedded system. Experiments were performed with five artificial intelligence algorithms. The work is innovative because it provides a temporal analysis of the driver's driving.

Keywords: Embedded Systems, Artificial intelligence, Dangerous driving, IoT

Sumário

1 Introdução	11
2 Fundamentação Teórica	13
2.1 Direção perigosa	13
2.2 Algoritmos de inteligência artificial	13
2.2.1 Algoritmos supervisionados	13
2.2.1.1 Árvores de Decisão	14
2.2.1.2 Random Forest	15
2.2.1.3 Máquinas de Vetores de Suporte (SVM)	15
2.2.1.4 K-NN	16
2.2.1.5 Redes Neurais Artificiais (ANN)	17
2.2.1.6 Long Short-Term Memory (LSTM)	17
2.2.2 Algoritmos de classificação	18
3 Trabalhos Relacionados	20
3.1 Tipos de dados utilizado	20
3.2 Base de dados utilizada	20
3.3 Algoritmos de inteligência artificial utilizados	21
3.4 Fonte da coleta de dados	21
3.5 Análise dos trabalhos relacionados	21
4 Descrição do sistema desenvolvido	23
4.1 Descrição geral do sistema	23

4.2 Implementação do sistema	24
4.2.1 Dispositivo de coleta	24
4.2.1.1 Módulo sensorial	25
4.2.1.2 Módulo de processamento	26
4.2.1.3 Módulo de comunicação	27
4.2.2 Servidor	27
4.2.2.1 Tratamento dos dados	27
4.2.2.2 Algoritmos de inteligência artificial	28
4.2.2.3 Implementação do servidor	28
4.2.3 Painel de controle	28
5 Resultados	30
6 Conclusão e trabalhos futuros	32

Capítulo 1

Introdução

Em 2022 o Brasil possuía uma frota ativa de aproximadamente 77.8 milhões de veículos[1].

Segundo o Registro Nacional de Acidentes e Estatísticas de Trânsito do Ministério dos Transportes[1], ocorreram 1.018.556 acidentes e 20.747 óbitos em acidentes de trânsito no Brasil em 2022. Desses, aproximadamente 65 mil acidentes ocorreram em rodovias federais brasileiras, onde 82% acarretaram vítimas ou mortes[3].

Além disso, os acidentes têm também acarretado prejuízos financeiros: o prejuízo causado por acidentes em rodovias brasileiras envolvendo caminhões gira em torno de R\$12 bilhões por ano[4].

De acordo com o Detran[5], 66% das causas de acidentes estão relacionadas a problemas com o condutor. Além disso, 55,7% desse total envolveram motoristas que realizaram uma ou mais ações de direção perigosa[6].

Atualmente já existem outros trabalhos na área de identificação perigosa, entretanto em sua maioria utilizam o telefone como dispositivo de coleta, o que pode facilitar que os dados coletados sejam adulterados pelo próprio usuário. Além disso, em nenhum dos estudos existe uma área para o acompanhamento da melhoria ou não da forma de direção dos motoristas.

Diante desse cenário, o objetivo deste trabalho é o desenvolvimento de um sistema de identificação de direção perigosa, composto por um sistema embarcado, um servidor e um painel de gestão. O sistema embarcado é composto por um módulo de sensoramento, um módulo de processamento, um módulo de comunicação e um módulo de feedback.

O módulo de sensoramento coleta dados de aceleração e velocidade angular utilizando o MPU-6050 e envia os dados para o módulo de processamento, que

realiza o pré-processamento dos dados e os envia para o módulo de comunicação. O módulo de comunicação envia os dados para o servidor. Foi escolhida a utilização do sistema embarcado para garantir a integridade dos dados coletados. O servidor analisa e identifica movimentos perigosos. Os dados coletados são armazenados e mostrados em um painel de gestão que pode ser usado para uma análise temporal das movimentações do veículo.

As maiores contribuições deste trabalho são a utilização de um sistema embarcado para garantir a integridade dos dados, além do painel de gestão, que poderá ser utilizado por empresas de transportes para que haja um melhor controle dos veículos que estejam trazendo mais riscos de acidentes e uma possível bonificação para os motoristas mais conscientes. Além disso, o sistema pode ser utilizado pelo Detran com os usuários que estejam fazendo curso de reciclagem e já foram anteriormente identificados com um perfil de direção perigosa, assim pode-se acompanhar a melhoria do condutor.

A estrutura do restante do trabalho será da seguinte forma: capítulo 2 irá abordar a fundamentação teórica, o capítulo 3 os trabalhos relacionados, o capítulo 4 a descrição do sistema, o capítulo 5 os resultados e o capítulo 6 as conclusões e trabalhos futuros.

Capítulo 2

Fundamentação teórica

Neste capítulo serão abordados tópicos que serão abordados durante esse trabalho. Entre eles estão: Direção perigosa, algoritmos de inteligência artificial e o sistema embarcado que foi utilizado.

2.1 Direção perigosa

De acordo com o Artigo 175 do código de trânsito brasileiro[2], a direção perigosa pode ser categorizada como a utilização do veículo para demonstrar ou exibir manobras perigosas, como arrancada brusca, derrapagem, frenagem ou arrastamento dos pneus. A infração é gravíssima, sofrendo penalidade de multa e até suspensão do direito de dirigir e apreensão do veículo. Além disso, o veículo pode ser apreendido e o documento de habilitação recolhido. Vale ressaltar que a consequência mais grave da direção perigosa é o aumento do risco de acidentes e óbitos.

2.2 Algoritmos de inteligência artificial

Neste trabalho, para a classificação do tipo de direção do usuário, serão utilizados algoritmos de inteligência artificial. Os algoritmos de inteligência artificial podem ser classificados em supervisionados ou não supervisionados.

2.2.1 Algoritmos supervisionados

Os algoritmos supervisionados são aqueles onde o modelo é treinado levando em consideração os resultados pré-definidos. Geralmente eles são utilizados para separar os dados em grupos já determinados, como por exemplo se a direção é perigosa ou defensiva. É comum informar que é utilizado um

“supervisor” para ajudar na classificação. Entre os algoritmos supervisionados estão: Árvores de Decisão, Random Forest, Máquinas de Vetores de Suporte (SVM) e Redes Neurais Artificiais (ANN)

2.2.1.1 Árvores de Decisão:

O algoritmo de árvores de decisão cria nós de decisão que são conectados a partir de regras pré definidas. Os nós são conectados em função de intervalos das características observadas, como pode ser visto na Figura 1.

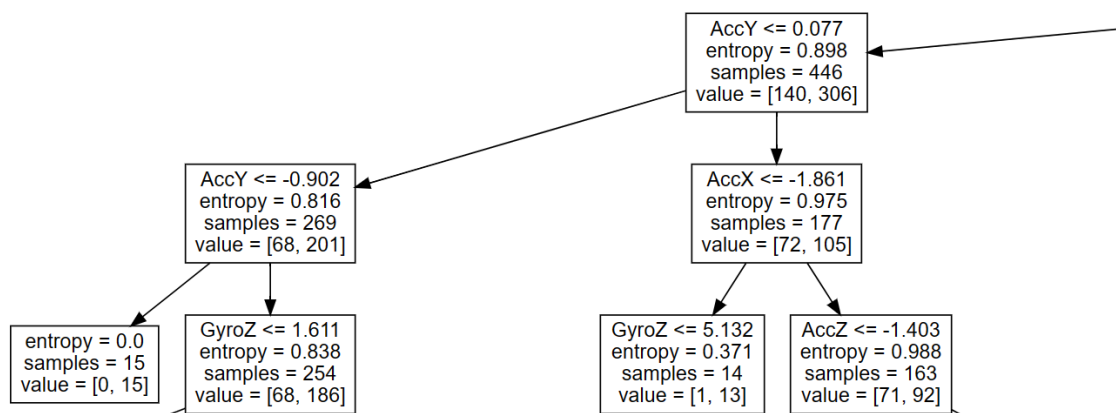


Figura 1 - Árvore de decisão

Dependendo da posição de cada nó na árvore de decisão mais preciso será o algoritmo e mais força uma característica vai ter em relação às outras. Para fazer essa escolha de influências é utilizado o ganho de informação. O ganho é calculado a partir da diferença da entropia do nó raiz e a média da entropia dos subnós. Na Figura 1, por exemplo, no nó $AccY \leq 0.077$ a entropia é 0.898 e a média da entropia de seus subnós é 0,8955. Assim, o ganho de informação do nó $AccY \leq 0.077$ é 0.0025. O nó raiz escolhido será o que possui o maior ganho de informação. Após escolher o nó raiz todas as outras entropias são recalculadas e continua-se a recalculando o ganho dos nós.

A entropia pode ser calculada da seguinte forma: $-\sum(p_i * \log_2(p_i))$, onde p_i é a proporção de exemplos pertencentes a uma classe específica, no caso da Figura 1 sendo representado por “samples” (no caso de $AccY$ seria 269).

O algoritmo de árvore de decisão é um dos mais conhecidos e um dos mais simples algoritmos de inteligência artificial e consegue trabalhar com várias categorias de classificação de forma mais simples. Além disso, o algoritmo consegue trabalhar com dados não normalizados e com dados fora do padrão. Entretanto com a árvore de decisão deve escolher que atributos têm mais influência, o algoritmo pode acabar não levando em consideração atributos também importantes para a análise.

2.2.1.2 Random Forest:

O algoritmo de random forest, ou floresta randômica, cria várias árvores de decisões distintas e aleatórias. A classificação final do algoritmo random forest será a classificação que mais se repete. Para a escolha dos ganhos de informação e da entropia, nesse caso, não são levadas em consideração todas as características mas sim apenas algumas (2 ou 3). Além disso, apenas algumas amostras da base de dados, diferente da árvore de decisão.

O algoritmo de random forest é um algoritmo mais preciso e tem menos chances de ter um overfitting, quando um modelo é treinado para se ajustar aos dados de treinamento, mas acaba sendo incapaz de generalizar com precisão para novos dados. Entretanto ela possui um custo operacional alto para treinar todas as árvores e gera um modelo que ocupa um espaço maior.

2.2.1.3 Máquinas de Vetores de Suporte (SVM)

Máquina de Vetores de Suporte ou Support Vectors Machine trabalha com o objetivo de identificar o melhor hiperplano para a separação das classes. A representação de hiperplanos pode ser visualizada na Figura 2. Os pontos mais próximos ao hiperplano são chamados de vetores de suporte. Para cada dado de entrada existirá uma dimensão no hiperplano. No caso deste trabalho será criado um hiperplano com 6 dimensões.

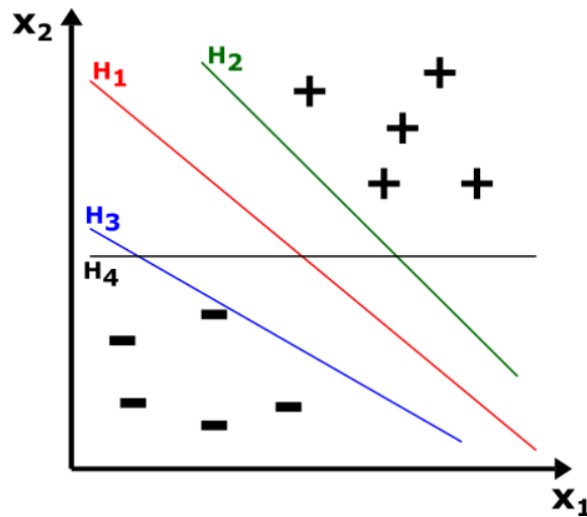


Figura 2 - Random Forest

2.2.1.4 K-NN:

O algoritmo de KNN ou K-nearest neighbors ou “K-vizinhos mais próximos”, é um algoritmo que utiliza a distância de um novo dado para os vizinhos já classificados. O dado será classificado com a mesma classificação dos dados com a menor distância do dado avaliado.

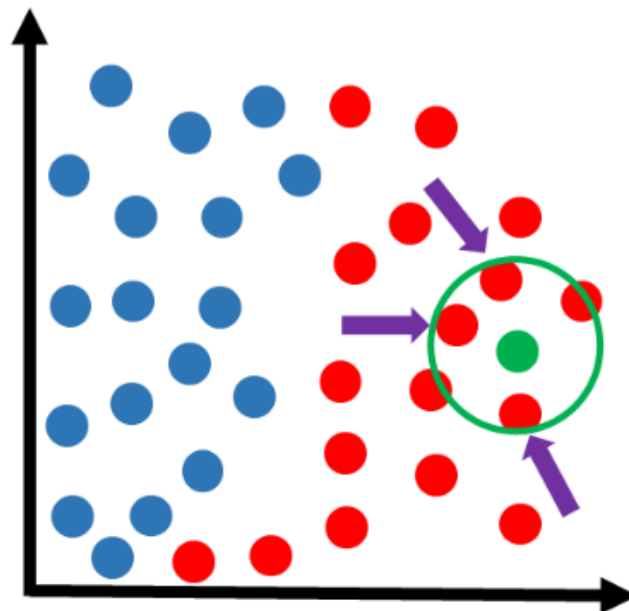


Figura 3 - KNN

Na Figura 3 o dado verde será classificado como vermelho, já que os pontos mais próximos a ele são vermelhos.

O algoritmo de KNN é um algoritmo simples e seu modelo consegue ser adaptado rapidamente a novos dados. Além disso, ele consegue ser adaptado a dados com várias características diferentes. Entretanto, como a cada inserção de dados o algoritmo calcula as distâncias dos pontos, pode ser computacionalmente custoso para uma base de dados grande. Além disso, como os dados são recalculados constantemente, a base de dados de treinamento deve ser salva, o que ocupa espaço da memória.

2.2.1.5 Redes Neurais Artificiais (ANN):

O algoritmo de ANN é composto por unidades de processamento chamadas de neurônios artificiais, ou perceptrons. Esses perceptrons são organizados em camadas interconectadas. Existem três camadas principais do algoritmo de ANN: Camada de Entrada, que recebe os dados de entrada, camadas ocultas, que são responsáveis por extrair características dos dados de entrada; camada de saída que produz a predição dos dados. A arquitetura de uma rede neurais artificiais podem ser vistos na Figura 4

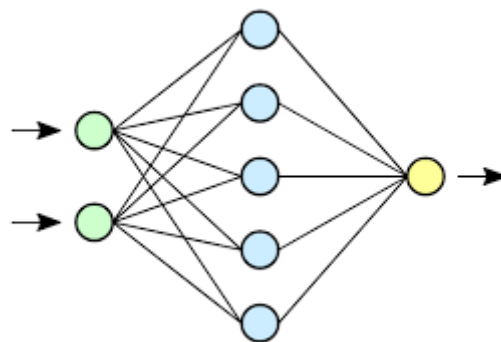


Figura 4 - Redes Neurais Artificiais

2.2.1.6 Long Short-Term Memory (LSTM):

O algoritmo de LSTM é um tipo de rede neural recorrente(RNN) desenvolvida para trabalhar com dados sequenciais, como reconhecimento de voz e séries temporais. A arquitetura de uma Rede Neural Recorrente é diferente dos

algoritmos anteriormente citados pois ele trabalha com um loop que consegue manter dados anteriores para uma análise futura, como pode ser visto na Figura 5.

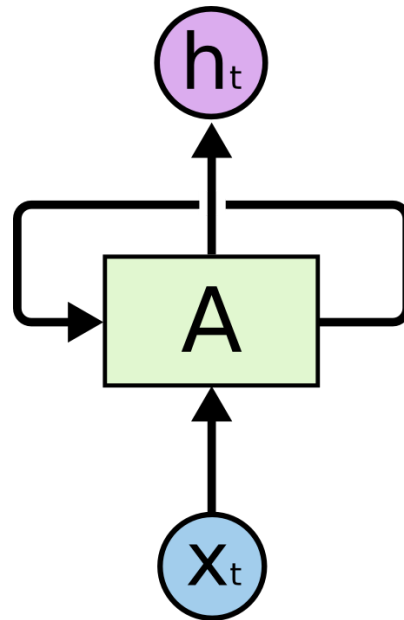


Figura 5 - LSTM

O LSTM é bastante utilizado para classificações supervisionadas, mas também pode ser usado em classificações não supervisionadas.

2.2.2 Algoritmos não supervisionados

Os algoritmos não supervisionados não precisam de classificações prévias. O sistema divide os dados em grupos com características semelhantes mas sem rotular as divisões. Um exemplo de algoritmo não supervisionado é o algoritmo de K-MEANS, que agrupa os dados em função da maior proximidade dos K, onde "K" representa o número de clusters desejados.

2.2.2 Algoritmos de classificação

Os algoritmos de classificação são algoritmos que são utilizados para atribuir uma categoria a um conjunto de dados baseado em suas características. Esses algoritmos possuem 3 etapas distintas:

Etapa de extração de características, que serão utilizadas para encontrar padrões nos dados, como pode ser visto um exemplo na Figura 6:

Amostras	É peludo?	Faz miau?	Tem penas?
amostra 1	1	1	0
amostra 2	0	0	1
amostra 3	1	1	0
amostra 4	1	0	0
amostra 5	0	1	0

Figura 6 -Etapa de extração de características

Etapa de classificação, que rotula cada grupo de dados, como pode ser visto na Figura 7:

Amostras	É peludo?	Faz miau?	Tem penas?	Tem asas?	É animal doméstico?	É pequeno?	Qual é o animal?
amostra 1	1	1	0	0	1	1	G
amostra 2	0	0	1	1	0	1	P
amostra 3	1	1	0	0	1	0	G
amostra 4	1	0	0	0	1	1	G
amostra 5	0	1	0	0	1	1	G

Figura 7 - Etapa de classificação

Etapa de criação do modelo aprendido: após rotular os dados é gerado um modelo de aprendizagem para que possa ser usado para o teste no futuro.

Capítulo 3

Trabalhos Relacionados

O tema de direção perigosa já é bastante abordado pela academia científica e com vários pontos de vista diferentes. Neste capítulo vão ser apresentados alguns trabalhos relacionados que abordam a identificação de direção perigosa em seus diferentes aspectos (tipo de dados utilizado, comparação de base de dados utilizada, algoritmos de inteligência artificial utilizados e implementação do sistema).

3.1 Tipos de dados utilizado

Diversos artigos atualmente abordam a utilização de visão computacional para identificação de direção perigosa. Os artigos [9] e [21] utilizam imagens captadas pela câmera para identificar direção perigosa.

Já os artigos [7] e [8] abordam a análise de dados de sensores IMU de seis eixos (giroscópio e acelerômetro) dentro do carro. Além disso, o artigo [11] também já abordou o uso do GPS para a análise de dados.

Os artigos [10], [22] trouxeram também uma abordagem mais embarcada no veículo onde se utiliza o CAN Bus.

Já o artigo [9] utilizou 9 dados diferentes: 6 eixos do sensor IMU, sensor de GPS e câmera.

3.2 Base de dados utilizada

Os trabalhos que abordam a identificação de direção perigosa através de inteligência artificial utilizam base de dados para algoritmos supervisionados para o treinamento dos algoritmos. O UAH-DriveSet [18] possui diversos parâmetros em sua base de dados para a análise classificatória, entre eles, 3 eixos de aceleração, 3 eixos de inclinação (giroscópio), velocidade do veículo e posição do GPS. Essa base de dados é amplamente utilizada por diversos artigos, entre eles os artigos [9], [15] e [21]. Essa base de dados possui 3 possíveis classificações: (normal, drowsy and aggressive).

Outra base de dados utilizada pelos artigos [7] e [20] é a Driving Behavior Dataset[19] que considera 3 eixos de inclinação(giroscópio) e 3 eixos de aceleração. Ele possui 3 possíveis classificações: AGGRESSIVE, NORMAL e SLOW.

3.3 Algoritmos de inteligência artificial utilizados

Os trabalhos que realizam a classificação e previsão de movimentos considerados perigosos usam algoritmos de inteligência artificial. O artigo [9] utiliza CNN para prever a atitude do veículo. Já o artigo [12] utiliza o algoritmo de KNN, enquanto os artigos [13] e [14] abordam o algoritmo KMEANS. Vários artigos abordam a aplicação do algoritmo de LSTM [7],[15],[16]. Enquanto isso, o artigo [17] utiliza algoritmo de SVM.

3.4 Fonte da coleta de dados

Entre os sistemas existentes, os smartphones foram o mais amplamente utilizados como pode ser lido nos artigos [7],[9], [11],[12],[15],[20],[21]. Alguns trabalhos falam sobre a utilização de microcomputadores nos veículos[8]. Outros utilizam microcontroladores [22]. Existem ainda artigos que utilizam dados coletados diretamente dos carros como In-Vehicle CAN Bus Data [10], [13] e [22] e também vehicle test data (VTD) [14].

3.5 Análise dos trabalhos relacionados

Analisando o tópico de tipos de dados utilizados, a identificação através de câmeras se torna muito custosa para uma avaliação em tempo real pois o processamento de imagens requer mais poder computacional de processamento do que o processamento de dados de IMU, CANBUS ou GPS. Além disso, a possibilidade de uma solução com o CANBUS se torna mais restrita. Segundo [7] a aplicação do GPS e consequentemente identificação da velocidade do veículo não é muito interessante pois o que define a direção perigosa é a forma que ele acelera ou freia em determinadas situações, não a sua velocidade nominal. Por exemplo: um carro pode estar na velocidade limite da via mas estar dirigindo de forma segura, enquanto outros veículos em uma velocidade mais baixa podem tomar decisões mais repentinas e mais

perigosas. Por esse motivo se escolheu o uso de dados de IMU de acelerômetro e giroscópio para a identificação de direção perigosa.

Já o tópico de base de dados utilizado, apesar do UAH-DRIVESET ser bastante utilizado, ele contempla não só valores de giroscópio e acelerômetro como também as gravações da rua. Por isso ele foi bastante utilizado por soluções que faziam um trabalho também de visão computacional. Os únicos dados de aceleração dessa base de dados não possuem nenhuma classificação, além de não existir uma métrica de como eles foram coletados. Como o trabalho proposto utiliza um sistema físico com os 6 eixos de liberdade do IMU, escolheu-se utilizar a base de dados Driving Behavior Dataset[19].

Como o Driving Behaviour Dataset[19] é uma base de dados já classificada, o mais adequado é utilizar um algoritmo da classe supervisionado. Entre os algoritmos possíveis estão: CNN, KNN, SVM, Decision Tree e Random Forest e Redes Neurais Artificiais (ANN). Além disso, o LSTM também pode ser considerado no contexto de direção perigosa, pois ele faz a análise de dados relacionados no tempo.

Para essa pesquisa foi necessário escolher uma fonte de coleta de dados para o sistema proposto. A utilização do celular, apesar de bem abrangente, não torna o dado muito confiável pois ele pode ser adulterado facilmente. Além disso, para utilizar o CAN Bus Data difere entre cada veículo. Por isso foi escolhido utilizar um microcomputador ou microcontrolador independente dentro do veículo.

Capítulo 4

Sistema desenvolvido

Neste capítulo será feita a descrição do sistema, tanto das suas características funcionais, qual é o fluxo do sistema e dos dados coletados, quanto da sua implementação técnica, com a descrição da implementação do sistema embarcado, do servidor e do painel de gestão.

4.1 Descrição geral do sistema

O sistema desenvolvido é composto por três subsistemas: um dispositivo de coleta, um servidor e um painel de gestão, como mostrado na Figura 8. O dispositivo de coleta, realiza a coleta dos dados de aceleração e velocidade angular do veículo e envia os dados para o servidor. No servidor os dados coletados são classificados por inteligência artificial como Agressivo ou Normal. Em função da resposta recebida o dispositivo de coleta envia um feedback luminoso ao motorista. Além disso, existe um painel de gestão onde o responsável consegue ter acesso ao tempo diário em que o motorista estava dirigindo de forma perigosa.

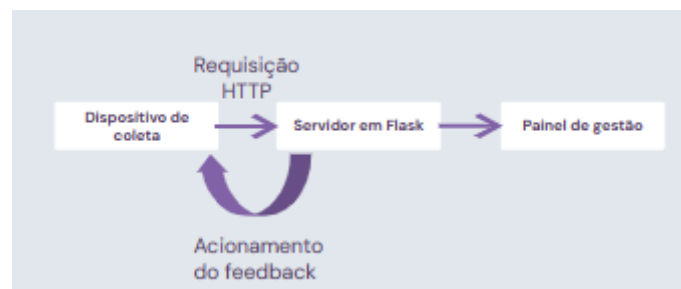


Figura 8 - Diagrama de blocos do sistema desenvolvido

4.2 Implementação do sistema

A seguir existe a descrição mais detalhada dos subsistemas.

4.2.1 Dispositivo de coleta

A implementação do dispositivo de coleta utilizado é mostrado na Figura 9 e seu protótipo pode ser visto na Figura 10. Ele é composto por um módulo de

sensoriamento, um módulo de processamento, um módulo de comunicação e um módulo de feedback. O módulo de sensoriamento coleta os dados de aceleração angular e aceleração e envia os dados para o módulo de processamento. O módulo de processamento, controlado por uma ESP-32, inicialmente faz a análise dos dados e identifica se o carro começou a se movimentar. Se o carro estiver em movimento ele envia os dados para o módulo de comunicação. O módulo de comunicação, também controlado pela ESP-32, envia os dados para o servidor que envia uma resposta para o dispositivo de coleta. Se o servidor identificar indícios de direção perigosa, o módulo de processamento envia os dados para o módulo de feedback que acende um led que pode ser visto pelo motorista.

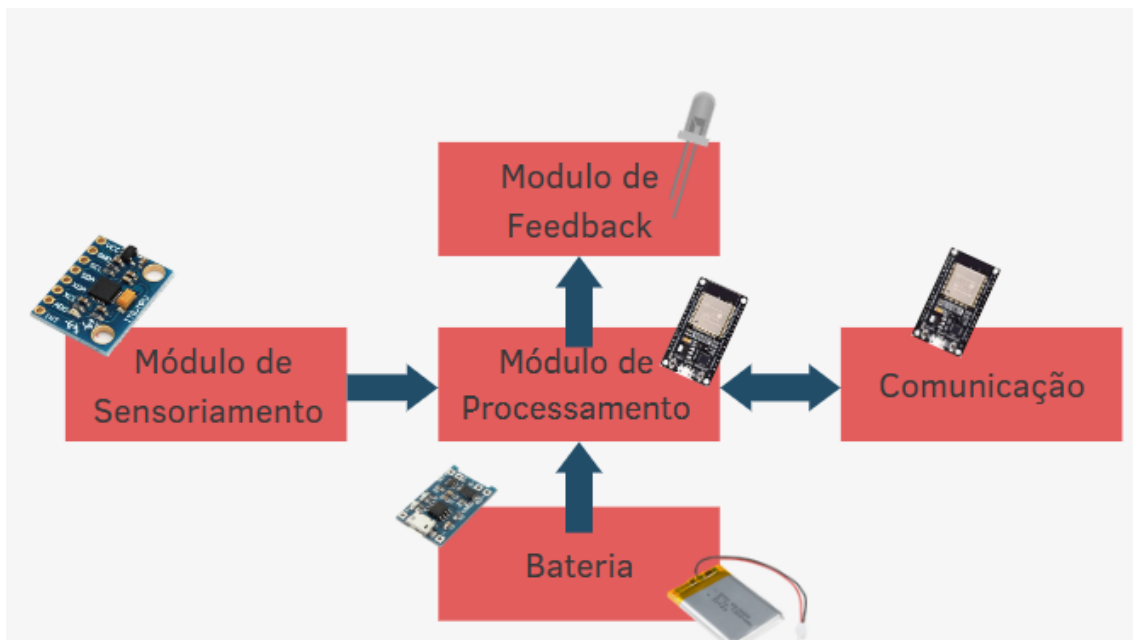


Figura 9 - Implementação do dispositivo de coleta

Para estruturar o dispositivo de coleta foi desenvolvido uma placa de circuito impresso, mostrada na Figura 10.

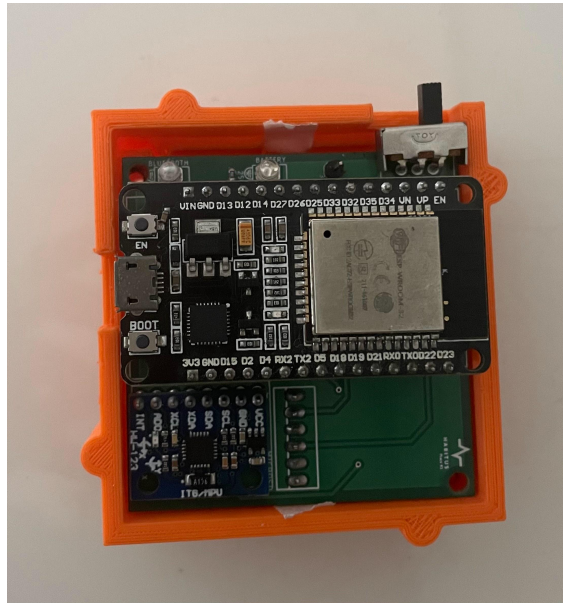


Figura 10 - Protótipo do sistema

4.2.1.1 Módulo de sensoriamento

O módulo de sensoriamento é responsável por medir a aceleração e a velocidade angular do veículo. Ele é composto por um sensor giroscópio e acelerômetro MPU-6050, que pode ser visto na Figura 11. O sensor capta os 3 eixos (x,y, z) de liberdade da velocidade angular em graus/s e os 3 eixos(x,y,z) de liberdade da aceleração em m/s^2 .

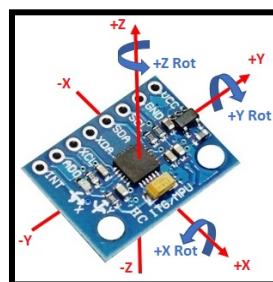


Figura 11 - MPU-6050

Os dados coletados são enviados para o módulo de processamento onde eles são pré-tratados e depois enviados para a nuvem para serem classificados. Os dados são coletados a cada 0.5s. A biblioteca utilizada é a Adafruit_MPU6050

para uma calibração automática dos sensores. Além disso, a biblioteca converte o sinal analógico em digital.

4.2.1.2 Módulo de processamento

O módulo de processamento, composto por uma ESP-32, é responsável por receber os dados do módulo sensorial, pré-processar os dados e identificar se o carro está em movimento e inicia o envio dos dados para o módulo de comunicação.

O primeiro passo do módulo de processamento, no setup, é identificar valores de erro dos sensores e definir valores fixos de ajustes que serão utilizados durante a coleta. Os valores de erro podem variar de sensor para sensor, por isso ele é calculado toda vez que a ESP-32 é reiniciada. Além disso, o módulo de processamento retira o valor da gravidade para que ela possa ser bem interpretada pelo servidor durante a classificação do tipo de direção do condutor do veículo.

Para a identificação da movimentação do carro foi utilizado um threshold para os seis eixos. Enquanto alguns dos valores lidos pelos 6 eixos do MPU-6050 se mantêm abaixo do threshold, o carro continua parado.

Ao iniciar o movimento do carro, o módulo de processamento começa a enviar a leitura dos dados do MPU-6050 para o módulo de comunicação que vai enviar os dados para o servidor. A máquina de estados do módulo de processamento pode ser vista na Figura 12. A lógica do módulo de processamento foi desenvolvida utilizando a plataforma ARDUINO IDE utilizando C/C++.

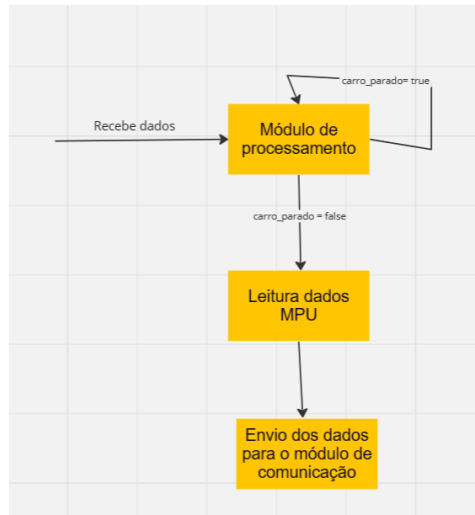


Figura 12 - Máquina de estados do módulo de processamento

4.2.1.3 Módulo de comunicação

No módulo de comunicação o dado pré-processado pelo módulo de processamento é enviado para o servidor. O dado é enviado via requisição HTTP pela ESP-32 utilizando a biblioteca WiFi.h. Os dados enviados são enviados em formato de JSON através de uma requisição POST. Na requisição HTTP são enviados os 6 eixos coletados pelo módulo de processamento. A resposta da requisição é enviada para o módulo de processamento e tratada. Se a resposta recebida for AGGRESSIVE o módulo de feedback através de um led vermelho é ativado. O módulo de processamento e o módulo de comunicação trabalham de forma sequencial.

4.2.2 Servidor

O servidor é responsável pela classificação do tipo de direção realizada. Caso o dispositivo de coleta esteja conectado à internet, o servidor a cada 0.5s recebe os 6 eixos de liberdade do MPU-6050. Com esses dados, o servidor trata os dados para serem então classificados pelo algoritmo de inteligência artificial.

4.2.2.1 Tratamento dos dados

Durante o tratamento dos dados recebidos pelo dispositivo de coleta, o servidor converte os valores recebidos em uma estrutura de dados tabular, transforma os dados em um array utilizando a biblioteca numpy. Além disso, os dados são

normalizados para que a comparação no algoritmo da IA seja imparcial em relação a um dos eixos.

4.2.2.2 Algoritmos de inteligência artificial

Para a classificação dos dados coletados pelo sistema físico foi escolhido fazer uma implementação em inteligência artificial para classificação dos dados. A base de dados já estava classificada em NORMAL, SLOW e AGGRESSIVE. Para a melhoria da acurácia dos modelos, as classificações NORMAL e SLOW foram transformadas em uma única classificação NORMAL. Utilizando essa base de dados, durante a pesquisa, foram treinados 6 algoritmos de classificação diferentes: Decision tree, Random Forest, Máquina de vetores de suporte(SVM), Redes Neurais Artificiais (ANN), LSTM e FeedForward Neural Network e testados utilizando o dataset de teste do Driving Behavior Dataset. Os resultados encontrados utilizando essa abordagem podem ser encontrados na seção de resultados.

A motivação para a escolha desses algoritmos foi baseada em algoritmos supervisionados, já que o Driving Behavior Dataset já possuía classificações prévias. Além disso o LSTM foi também analisado já que os dados possuíam interligação temporal, já que apesar do dado ser lido a cada 0.5s uma direção perigosa pode ser identificada em aproximadamente 10s [9].

Após o treinamento do algoritmo de inteligência artificial escolhido, um modelo foi gerado e adicionado ao código do servidor. Dessa forma, a cada leitura do dispositivo de coleta os dados são alimentados e classificados pelo modelo pré treinado. Após a classificação pelo servidor, foi utilizado um Threshold para identificar se um conjunto de leituras deve ou não ser classificado como agressivo. Foi escolhido o valor de 80% arbitrariamente. Se durante a classificação for identificada uma movimentação agressiva os dados são enviados tanto para o dispositivo de coleta quanto para o painel de controle.

4.2.2.3 Implementação do servidor

O servidor foi implementado em Python utilizando a biblioteca Flash[24]. Para que ele conseguisse se comunicar com o dispositivo de coleta foi necessário criar uma hospedagem provisória. Para isso foi utilizado o flask_ngrok, uma extensão do Flask que permite a obtenção de um URL público temporário para o ser utilizado pelo servidor.

4.2.3 Painel de controle

Além do sistema embarcado e a inteligência artificial, também foi desenvolvido um painel de gestão para o controle do tempo por dia que foi executado um movimento perigoso. Para isso, foi criado um painel de gestão utilizando o GRAFANA [25], como pode ser visto na Figura 13.

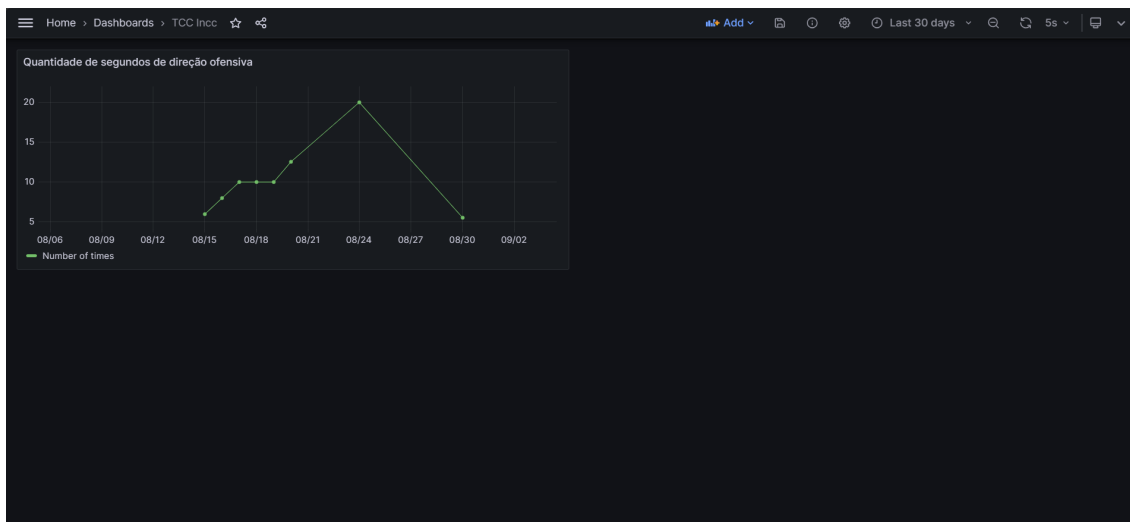


Figura 13 - Painel de gestão

Para a conexão entre o servidor e o grafana foi utilizado um google sheets que salva a quantidade de segundos lidos e a data.

Capítulo 5

Resultados

Para validar os algoritmos de inteligência artificial foi utilizada a base de dados Driving Behavior Dataset[19] que considera 3 eixos de inclinação(giroscópio) e 3 eixos de aceleração. Ele possui 3 possíveis classificações: AGGRESSIVE, NORMAL e SLOW. Para melhorar a análise foi considerado apenas dois tipos de classificação: AGGRESSIVE e NORMAL. Utilizou-se uma base de dados para treino e uma base de dados para testes. Antes do treinamento dos dados foi feita uma limpeza nos dados para retirar ruídos gerados durante a coleta. Para a validação do sistema foram testados 5 algoritmos de inteligência artificial, onde a base de dados de treinamento do Driving Behavior Dataset[19] foi utilizado. A acurácia de cada um dos algoritmos avaliados pode ser analisada na Tabela 1.

Tipo do algoritmo de IA	Acurácia
Decision tree	64,62%
Random Forest	74,01%
Máquina de vetores de suporte(SVM)	76,03%
Redes Neurais Artificiais (ANN)	72.78%
LSTM	76,6%
FeedForward Neural Network	76,09%

Tabela 1 - Acurácia dos algoritmos de IA

Entre os algoritmos analisados a Máquina de vetores de suporte(SVM), LSTM e FeedForward Neural Network tiveram aproximadamente a mesma acurácia. A partir da primeira análise deve-se então ser analisada a resposta do sistema aos dados coletados pelo sistema embarcado. Foi escolhido o algoritmo de

Máquina de Vetores de Suporte(SVM) pois ele não precisava de um envio grande de dados para fazer a classificação e consequentemente não afetava a memória da ESP-32.

Utilizando o algoritmo SVM, os dados eram lidos pelo MPU-6050, normalizados e classificados pelo servidor. A validação do sistema físico foi realizada de forma simulada, movimentando o sistema embarcado como se fosse o movimento do carro. As respostas do sistema não eram muito precisas e só acusavam se o movimento tivesse sido realmente muito brusco. A resposta mais fidedigna foi no eixo Z, muitas vezes utilizado para a identificação de buracos nas vias. O tempo de resposta da classificação era em torno de 5 segundos, o que deixa a simulação não muito fidedigna.

Além disso, o sistema criado tem algumas limitações para ser usado na vida real. O sistema se conecta ao servidor através de requisições HTTP, o que necessita a conectividade a internet em todos os momentos. Além disso, a base de dados utilizada para o treinamento da inteligência artificial foi coletada com dados do giroscópio e acelerômetro do celular, o que pode trazer discrepâncias para a análise dos dados do sistema embarcado.

Durante a coleta de dados da Driving Behavior Dataset[19] o percurso da Figura 14 foi percorrido três vezes(uma vez de modo agressivo, uma vez de modo normal e uma vez de modo lento). Entretanto, nos dados coletados no modo agressivo foi identificado dados que poderiam ser classificados como normais, pois o usuário não conduziu o veículo de forma perigosa 100% do tempo. Isso pode afetar a acurácia do algoritmo com os dados coletados pelo sistema embarcado, já que os dados estão sendo classificados individualmente e não leva em consideração o tempo.

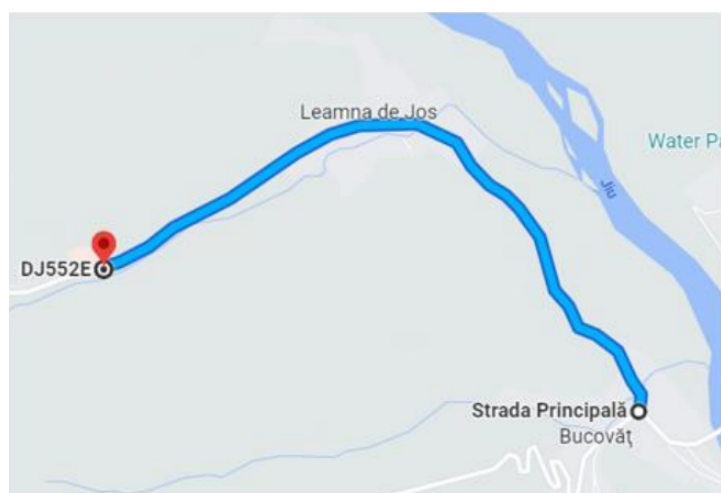


Figura 14

Capítulo 6

Conclusão e trabalhos futuros

Foi desenvolvido neste trabalho sistema de identificação de direção perigosa, que coleta dados de aceleração e velocidade angular utilizando o MPU-6050, analisa e identifica em tempo real movimentos perigosos utilizando inteligência artificial. Também foi desenvolvido um painel de gestão, para realização de análise temporal para uma futura tomada de decisões, que indica quanto tempo por dia o usuário realizou movimentações perigosas.

O sistema desenvolvido conseguiu atingir os resultados esperados e entregar um sistema funcional em testes de laboratório. Entretanto, para que o sistema funcione com usuários reais algumas implementações futuras devem ser desenvolvidas. Entre elas, a acurácia da inteligência artificial, tornar a classificação do sistema realmente em tempo real, sistema funcionar em locais remotos, além de melhorias para trazer melhores análises para o sistema.

Para o sistema funcionar em locais remotos, pode ser implementada uma rede LPWAN como Lora para não depender de Wifi. Para um trabalho futuro deve-se desenvolver um sistema para salvar os dados localmente em caso de falha na conexão com o servidor.

Para a melhoria da acurácia deve ser coletada uma nova base de dados própria utilizando o hardware desenvolvido. Além disso, os pontos de identificação NORMAL ou AGRESSIVE seriam identificados por trecho de via e não o percurso completo, já que nem todo motorista dirige 100% agressivo ou 100% normal.

Para tornar a classificação do sistema em tempo real deve-se implementar a solução para funcionar embarcada em um microcomputador que pode suportar o processamento necessário. Entre as possibilidades estaria o Raspberry Pi. A utilização do Raspberry Pi também ajudaria na necessidade de conexão com a internet durante todo o caminho, pensando que em várias partes das estradas brasileiras a conectividade é muito precária. Assim, o processamento dos dados e os feedbacks seriam feitos em tempo real e os dados seriam enviados para a nuvem só quando tivesse conexão à internet.

Entre possíveis implementações de novas funcionalidades estão: identificar derrapagem na pista durante épocas de chuva e neve utilizando um sensor auxiliar. Identificar locais de maior risco para reparação da via ao implementar a utilização do gps no sistema.

Capítulo 7

Referências

- [1] [Registro Nacional de Acidentes e Estatísticas de Trânsito — Ministério dos Transportes \(www.gov.br\)](http://www.gov.br)
- [2] <https://www.ctbdigital.com.br/artigo/art175>
- [3] <https://autoesporte.globo.com/servicos/noticia/2023/02/pesquisa-mostra-o-perfil-e-idade-do-motorista-que-se-envolve-em-acidentes-nas-estradas-no-brasil.g.html>
- [4] <https://www.ontruck.com.br/noticia/10614/acidentes-nas-estradas-causam-pr-ejuizos-de-mais-de-r-12-bilhoes-por-ano>
- [5] <https://www.detran.pr.gov.br/arquivos/File/habilitacao/manualdehabilitacao/manualdehabparte6.pdf>
- [6] [Aggressive Driving Research Update 2009 \(dot.gov\)](#)
- [7] Cojocar, Ion, Paul-Stefan Popescu, and Marian Cristian Mihaescu. "Driver Behaviour Analysis based on Deep Learning Algorithms." *RoCHI*. 2022.
- [8] J. Huang, Z. Huang and K. Chen, "Combining low-cost Inertial Measurement Unit (IMU) and deep learning algorithm for predicting vehicle attitude," 2017 IEEE Conference on Dependable and Secure Computing, Taipei, Taiwan, 2017, pp. 237-239, doi: 10.1109/DESEC.2017.8073847.
- [9] K. Saleh, M. Hossny and S. Nahavandi, "Driving behavior classification based on sensor data fusion using LSTM recurrent neural networks," 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 2017, pp. 1-6, doi: 10.1109/ITSC.2017.8317835.
- [10] W. -H. Chen, Y. -C. Lin and W. -H. Chen, "Comparisons of Machine Learning Algorithms for Driving Behavior Recognition Using In-Vehicle CAN Bus Data," 2019 Joint 8th International Conference on Informatics, Electronics &

Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Spokane, WA, USA, 2019, pp. 268-273, doi: 10.1109/ICIEV.2019.8858531.

[11] Guo, J.; Liu, Y.; Zhang, L.; Wang, Y. Driving Behaviour Style Study with a Hybrid Deep Learning Framework Based on GPS Data. *Sustainability* **2018**, *10*, 2351. <https://doi.org/10.3390/su10072351>

[12] Christy, A., A. Jesudoss, and M. S. Roobini. "Driver Helper: A Mobile Based Application to Predict Driver Behaviour Using Classification Techniques." *Int. J. Adv. Sci. Technol* 29.7s (2020): 4429-4438.

[13]U. Fugiglando et al., "Driving Behavior Analysis through CAN Bus Data in an Uncontrolled Environment," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 2, pp. 737-748, Feb. 2019, doi: 10.1109/TITS.2018.2836308.

[14]B. Shi et al., "Evaluating Driving Styles by Normalizing Driving Behavior Based on Personalized Driver Modeling," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45, no. 12, pp. 1502-1508, Dec. 2015, doi: 10.1109/TSMC.2015.2417837.

[15] K. Saleh, M. Hossny and S. Nahavandi, "Driving behavior classification based on sensor data fusion using LSTM recurrent neural networks," 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 2017, pp. 1-6, doi: 10.1109/ITSC.2017.8317835.

[16]Zou, Yajie, et al. "Vehicle acceleration prediction based on machine learning models and driving behavior analysis." *applied sciences* 12.10 (2022): 5259.

[17]Karri, S.L., De Silva, L.C., Lai, D.T.C. *et al.* Classification and Prediction of Driving Behaviour at a Traffic Intersection Using SVM and KNN. *SN COMPUT. SCI.* 2, 209 (2021). <https://doi.org/10.1007/s42979-021-00588-7>

[18] E. Romera, L. M. Bergasa and R. Arroyo, "Need data for driver behaviour analysis? Presenting the public UAH-DriveSet," 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 2016, pp. 387-392, doi: 10.1109/ITSC.2016.7795584.

[19] Paul-Stefan Popescu, & Cojocaru Ion. (2022). <i>Driving Behavior</i> [Data set]. Kaggle. <https://doi.org/10.34740/KAGGLE/DSV/3748585>

[20] Kanwal, Khadija, et al. "Smartphone Inertial Measurement Unit Data Features for Analyzing Driver Driving Behavior." *IEEE Sensors Journal* (2023).

[21] Moukafih, Youness, Hakim Hafidi, and Mounir Ghogho. "Aggressive driving detection using deep learning-based time series classification." *2019 IEEE international symposium on INnovations in intelligent SysTems and applications (INISTA)*. IEEE, 2019.

[22] Mohammed, Karrouchi, et al. "Intelligent driver monitoring system: An Internet of Things-based system for tracking and identifying the driving behavior." *Computer Standards & Interfaces* 84 (2023): 103704

[23] Dijksterhuis, Chris, et al. "The impact of immediate or delayed feedback on driving behaviour in a simulated Pay-As-You-Drive system." *Accident Analysis & Prevention* 75 (2015): 93-104.

[24] <https://flask.palletsprojects.com/en/2.3.x/>

[25] <https://grafana.com/>

Assinaturas

Recife, 22 de setembro de 2023

Luisa Nogueira Costa Cavalcante

(Aluno)

Adriano Augusto de Moraes Sarmento

(Orientador)