



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Jeferson Menegazzo

**Percepção Veicular Adaptativa Baseada em Sensoriamento Inercial e Deep Learning**

Florianópolis

2021



Jeferson Menegazzo

## **Percepção Veicular Adaptativa Baseada em Sensoriamento Inercial e Deep Learning**

Dissertação submetida ao Programa de Pós-Graduação em Ciência da Computação para a obtenção do título de mestre em Ciência da Computação.

Orientador: Prof. Aldo von Wangenheim, Dr. rer. nat.

Florianópolis

2021

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Menegazzo, Jeferson

Percepção Veicular Adaptativa Baseada em Sensoriamento  
Inercial e Deep Learning / Jeferson Menegazzo ;  
orientador, Aldo von Wangenheim, 2021.  
157 p.

Dissertação (mestrado) - Universidade Federal de Santa  
Catarina, Centro Tecnológico, Programa de Pós-Graduação em  
Ciência da Computação, Florianópolis, 2021.

Inclui referências.

1. Ciência da Computação. 2. Sistemas de Transporte  
Inteligentes. 3. Classificação de Tipo de Superfície de  
Pista. 4. Classificação de Qualidade de Superfície de Pista.  
5. Reconhecimento de Lombadas. I. von Wangenheim, Aldo.  
II. Universidade Federal de Santa Catarina. Programa de Pós  
Graduação em Ciência da Computação. III. Título.

Jeferson Menegazzo

**Percepção Veicular Adaptativa Baseada em Sensoriamento Inercial e Deep Learning**

O presente trabalho em nível de mestrado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Fábio Alexandrini, Dr.  
Instituto Federal Catarinense

Prof. Alexandre Gonçalves Silva, Dr.  
Universidade Federal de Santa Catarina

Prof. Mateus Grellert da Silva, Dr.  
Universidade Federal de Santa Catarina

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de mestre em Ciência da Computação.

---

Prof. Vania Bogorny, Dr<sup>a</sup>  
Coordenadora do Programa

---

Prof. Aldo von Wangenheim, Dr. rer. nat.  
Orientador

Florianópolis, 2021.



## **AGRADECIMENTOS**

Agradeço aos meus pais, Benvenuto Menegazzo Filho e Selma Aparecida Matos de Andrade Menegazzo, pelo carinho e apoio de sempre.

À minha irmã Jéssica Menegazzo, que sempre me motivou e aconselhou, apoiando em todos os momentos.

Ao meu orientador, professor Aldo von Wangenheim, pelas oportunidades e contribuições para realização deste trabalho.





## RESUMO

Os sistemas de transporte se estabeleceram ao longo da história como um dos principais condicionantes ao desenvolvimento humano. Com o avanço das tecnologias computacionais, surgiram os sistemas de transporte inteligentes, nos quais sensores são empregados na infraestrutura de transporte e seus participantes, de forma a gerar dados brutos que, processados por modelos de IA, geram informações situacionais acerca do modal. Neste contexto, foram desenvolvidas diversas tecnologias de abordagem ativa e passiva. Dentre as tecnologias passivas, existem as baseadas em vibração, realizadas através de sensores inerciais, as quais podem gerar informações na forma de percepções veiculares, de forma segura, não poluente e de baixo custo. Entretanto, ao contrário de áreas como a visão computacional, o sensoriamento inercial tem sido pouco explorado, onde as soluções propostas na literatura não são adaptáveis para ampla aplicação em cenários do mundo real, se configurando normalmente como prova de conceito através modelos simples. Neste sentido, dada a diversidade contextual na qual a solução pode ser submetida, existem diversos fatores de dependência que interferem e influenciam os valores dos sinais amostrados com estes sensores, de forma que a adaptabilidade da solução a estes fatores se mostra um requisito essencial para prover confiabilidade e, por sua vez, possibilitar uma ampla aplicação. Com este objetivo, neste trabalho foi proposto o desenvolvimento de modelos de percepções veiculares baseados em sinais de sensores inerciais, capazes de operar de forma confiável em variações contextuais relacionados aos fatores de dependência: diferentes veículos, estilos de condução e ambientes. Neste trabalho focou-se no desenvolvimento das percepções de tipo de superfície de pista, qualidade de superfície de pista, e detecção de lombadas. Para o desenvolvimento e validação dos modelos, foram coletados nove conjuntos de dados com variações contextuais, utilizando três modelos de veículos, com três motoristas, em três ambientes distintos, nos quais existem três tipos de superfície, além de variações no estado de conservação e a presença de obstáculos e irregularidades. Os dados coletados foram utilizados em experimentos para avaliar aspectos como a influência do ponto de coleta de dados do veículo, o domínio de análise, as características de entrada do modelo e a janela de dados. Posteriormente, foi avaliada a capacidade de generalização do aprendizado dos modelos para contextos desconhecidos, ou seja, seu comportamento quando aplicado a dados amostrados em um veículo, motorista ou ambiente desconhecido, analisando assim sua adaptabilidade. Os experimentos foram realizados com modelos baseados em aprendizado de máquina clássico e *deep learning*, onde o melhor modelo para classificação de tipo de superfície foi uma rede CNN, a qual classificou segmentos de terra, paralelepípedo e asfalto com acurácia média de 92,70%; o melhor modelo para classificação de qualidade foi uma rede CNN, a qual classificou segmentos nos níveis bom, regular e ruim com acurácia média de 93,52%; e o melhor modelo para reconhecimento de lombadas foi uma rede híbrida CNN-LSTM, a qual detectou lombadas com acurácia média de 98,59%.

**Palavras-chave:** Classificação de Tipo de Superfície de Pista. Classificação de Qualidade de Superfície de Pista. Detecção de Lombadas. Sensores Inerciais. Aprendizado Profundo.



## ABSTRACT

Transport systems have been established throughout history as one of the main constraints to human development. With the advancement of computational technologies, intelligent transport systems emerged, in which sensors are used in the transport infrastructure and its participants, in order to generate raw data that, processed by AI models, generate situational information about the modal. In this context, several technologies with an active and passive approach have been developed. Among the passive technologies, there are those based on vibration, performed through inertial sensors, which can generate information in the form of vehicle perceptions, in a safe, non-polluting and low-cost way. However, in contrast to areas such as computer vision, inertial sensing has been little explored, where the solutions proposed in the literature are not adaptable for wide application in real-world scenarios, usually configuring themselves as proof of concept through simple models. In this way, given the contextual diversity in which the solution can be submitted, there are several dependency factors that interfere and influence the values of the signals sampled with these sensors, so that the adaptability of the solution to these factors is an essential requirement to provide reliability and, in turn, enable a wide application. With this objective in mind, in this work proposes the development of vehicle perception models based on inertial sensor signals, capable of operating reliably in contextual variations related to dependency factors: different vehicles, driving styles and environments. In this work focused on the development of perceptions of road surface type, road surface quality, and speed bump detection. For the development and validation of the models, nine datasets with contextual variations were collected, using three vehicle models, with three different drivers, in three different environments, in which there are three different surface types, in addition to variations in the conservation state and the presence of obstacles and irregularities. The collected data were used in experiments to evaluate aspects such as the influence of the vehicle's data collection point, the analysis domain, the model's input features and the data window size. Subsequently, the models' ability to generalize their learning to unknown contexts was evaluated, i.e., their behavior when applied to data sampled in an unknown vehicle, driver or environment, thus analyzing their adaptability. The experiments were carried out with models based on classical and deep machine learning, where the best model for road surface type classification was a CNN network, which classified dirt, cobblestone and asphalt segments with an average accuracy of 92.70%; the best model for road quality classification was a CNN network, which classified segments into good, fair and poor levels with an average accuracy of 93.52%; and the best model for speed bump recognition was a hybrid CNN-LSTM network, which detected speed bumps with an average accuracy of 98.59%.

**Palavras-chave:** Road Surface Type Classification. Road Surface Quality Classification. Speed Bump Detection. Inertial Sensors. Deep Learning.



## LISTA DE ILUSTRAÇÕES

Figura 1 – Abordagens de sensoriamento em ITS . . . . .	22
Figura 2 – Percepções veiculares produzidas através de dados de sensores inerciais . . . . .	23
Figura 3 – O modelo <i>Quarter Car</i> . . . . .	34
Figura 4 – Execução da técnica KMC . . . . .	36
Figura 5 – Execução da técnica KNN . . . . .	37
Figura 6 – Execução da técnica SVM . . . . .	38
Figura 7 – Célula de uma LSTM com <i>auto-loops</i> . . . . .	39
Figura 8 – Componentes de uma célula de LSTM . . . . .	40
Figura 9 – Componentes de uma célula de GRU . . . . .	41
Figura 10 – Convolução em dados 1D . . . . .	42
Figura 11 – Convolução em dados 3D . . . . .	42
Figura 12 – Matriz confusão de classificação . . . . .	43
Figura 13 – Referenciais: (a) Referencial do sensor. (b) Referencial do sensor interno a dispositivos móveis. (c) Referencial da Terra. (d) Referencial do veículo. . . . .	49
Figura 14 – Sensores inerciais utilizados nos estudos revisados . . . . .	53
Figura 15 – Plataformas utilizadas nos estudos revisados . . . . .	53
Figura 16 – Colocação dos sensores no veículo . . . . .	54
Figura 17 – Métodos utilizados no pré-processamento de sinais dos estudos revisados . . . . .	56
Figura 18 – Domínios de análise utilizados nos estudos revisados . . . . .	56
Figura 19 – Técnicas utilizadas para reconhecimento e classificação de padrões de percepção veicular . . . . .	57
Figura 20 – Exterocepções encontradas nos estudos analisados . . . . .	61
Figura 21 – Propriocepções encontradas nos estudos analisados . . . . .	61
Figura 22 – Colocação e posicionamento da rede de sensores no veículo . . . . .	79
Figura 23 – Tipos de superfície presentes nos cenários . . . . .	81
Figura 24 – Mapa das superfícies presentes nos cenários. . . . .	81
Figura 25 – Lombadas em diferentes superfícies presentes nos cenários . . . . .	82
Figura 26 – Lombadas em diferentes superfícies presentes nos cenários . . . . .	83
Figura 27 – Processo de GT de anotação automatizada por máquina para qualidade de superfície . . . . .	85
Figura 28 – Diferentes níveis de qualidade de superfície presentes nos cenários . . . . .	85
Figura 29 – Divisão dos conjuntos de dados PVS em grupos de treinamento e validação . . . . .	89
Figura 30 – Combinação de tipos de experimentos . . . . .	89
Figura 31 – Melhor modelo de LSTM para classificação de superfície de pista . . . . .	92
Figura 32 – Melhor modelo de CNN para classificação do tipo de superfície de pista . . . . .	93

Figura 33 – Melhor modelo de LSTM-CNN para classificação do tipo de superfície de pista . . . . .	94
Figura 34 – Matriz de confusão para cada melhor modelo de técnicas clássicas de aprendizado de máquina . . . . .	97
Figura 35 – Matriz de confusão para cada melhor modelo de técnicas de <i>deep learning</i> .	100
Figura 36 – Combinação de tipos de experimentos . . . . .	105
Figura 37 – Modelo LSTM para classificação do tipo de superfície de pista . . . . .	106
Figura 38 – Modelo GRU para classificação do tipo de superfície de pista . . . . .	106
Figura 39 – Modelo CNN para classificação do tipo de superfície de pista . . . . .	107
Figura 40 – Matriz de confusão para o modelo de CNN em cada ponto de coleta de dados	111
Figura 41 – Combinação de tipos de experimentos . . . . .	115
Figura 42 – Modelo LSTM para classificação de qualidade de superfície de pista . . . .	116
Figura 43 – Modelo GRU para classificação de qualidade de superfície de pista . . . .	117
Figura 44 – Modelo CNN para classificação de qualidade de superfície de pista . . . .	117
Figura 45 – Modelo CNN-LSTM para classificação de qualidade de superfície de pista .	118
Figura 46 – Modelo ConvLSTM para classificação de qualidade de superfície de pista .	119
Figura 47 – Matriz de confusão para o modelo CNN em cada ponto de coleta de dados .	124
Figura 48 – Modelo LSTM para detecção de lombadas . . . . .	128
Figura 49 – Modelo GRU para detecção de lombadas . . . . .	129
Figura 50 – Modelo CNN para detecção de lombadas . . . . .	130
Figura 51 – Modelo CNN-LSTM para detecção de lombadas . . . . .	130
Figura 52 – Modelo ConvLSTM para detecção de lombadas . . . . .	131
Figura 53 – Matriz de confusão para o modelo CNN-LSTM em cada ponto de coleta de dados . . . . .	136

## LISTA DE TABELAS

Tabela 1 – Os parâmetros <i>Golden Car</i> . . . . .	35
Tabela 2 – Definições da busca . . . . .	46
Tabela 3 – <i>Strings</i> de busca para exterocepção veicular . . . . .	47
Tabela 4 – <i>Strings</i> de busca para propriocepção veicular . . . . .	47
Tabela 5 – Resultados sumarizados das buscas . . . . .	48
Tabela 6 – Referenciais para os dados de aceleração . . . . .	49
Tabela 7 – Referenciais para os dados de taxa de rotação . . . . .	50
Tabela 8 – Comparação de conjuntos de dados públicos . . . . .	77
Tabela 9 – Hardware da rede de sensores . . . . .	78
Tabela 10 – Conjuntos de dados produzidos . . . . .	80
Tabela 11 – Arquivos que compõe cada conjunto de dados . . . . .	80
Tabela 12 – Quantificação para as classes de dados de tipo de superfície . . . . .	82
Tabela 13 – Quantificação para as classes de dados de lombadas . . . . .	83
Tabela 14 – Quantificação para as classes de dados de qualidade de superfície . . . . .	86
Tabela 15 – Distribuição de classes de dados de tipo de superfície de pista . . . . .	90
Tabela 16 – Modelos de LSTM para classificação do tipo de superfície de pista . . . . .	92
Tabela 17 – Modelos de CNN para classificação do tipo de superfície de pista . . . . .	93
Tabela 18 – Modelos de CNN-LSTM para classificação do tipo de superfície de pista . . . . .	94
Tabela 19 – Média de acurácia em validação para os modelos de KMC . . . . .	95
Tabela 20 – Média de acurácia em validação para os modelos de SVM . . . . .	95
Tabela 21 – Média de acurácia em validação para os modelos de KNN . . . . .	96
Tabela 22 – Valores de acurácia para os melhores modelos baseados em técnicas clássicas de aprendizado de máquina . . . . .	97
Tabela 23 – Métricas de avaliação para cada melhor modelo baseado em técnicas clássicas de aprendizado de máquina . . . . .	98
Tabela 24 – Média de acurácia em validação para redes baseadas em LSTM . . . . .	98
Tabela 25 – Média de acurácia em validação para redes baseadas em CNN . . . . .	99
Tabela 26 – Média de acurácia em validação para redes baseadas em CNN-LSTM . . . . .	99
Tabela 27 – Valores de acurácia para os melhores modelos baseados em técnicas <i>deep learning</i> . . . . .	100
Tabela 28 – Métricas de avaliação para cada melhor modelo baseado em técnicas de <i>deep learning</i> . . . . .	101
Tabela 29 – Média de acurácia para colocação próximo e abaixo da suspensão . . . . .	108
Tabela 30 – Média de acurácia para colocação próximo e acima da suspensão . . . . .	108
Tabela 31 – Média de acurácia para colocação no painel de controle . . . . .	109

Tabela 32 – Valores de acurácia para o modelo baseado em CNN . . . . .	110
Tabela 33 – Métricas de avaliação para o modelo baseado em CNN . . . . .	111
Tabela 34 – Distribuição de classes de dados de qualidade de superfície de pista . . . . .	115
Tabela 35 – Valores de acurácia em validação obtidos pelo modelo LSTM . . . . .	120
Tabela 36 – Valores de acurácia em validação obtidos pelo modelo GRU . . . . .	120
Tabela 37 – Valores de acurácia em validação obtidos pelo modelo CNN . . . . .	121
Tabela 38 – Valores de acurácia em validação obtidos pelo modelo CNN-LSTM . . . . .	121
Tabela 39 – Valores de acurácia em validação obtidos pelo modelo ConvLSTM . . . . .	122
Tabela 40 – Métricas de avaliação para a melhor configuração de cada modelo DNN . . . . .	123
Tabela 41 – Distribuição de classes de lombadas . . . . .	127
Tabela 42 – Distribuição de classes de lombadas após <i>downsampling</i> . . . . .	127
Tabela 43 – Valores de acurácia em validação obtidos pelo modelo LSTM . . . . .	132
Tabela 44 – Valores de acurácia em validação obtidos pelo modelo GRU . . . . .	132
Tabela 45 – Valores de acurácia em validação obtidos pelo modelo CNN . . . . .	133
Tabela 46 – Valores de acurácia em validação obtidos pelo modelo CNN-LSTM . . . . .	133
Tabela 47 – Valores de acurácia em validação obtidos pelo modelo ConvLSTM . . . . .	134
Tabela 48 – Métricas de avaliação para a melhor configuração de cada modelo DNN . . . . .	135



## LISTA DE ABREVIATURAS E SIGLAS

ADAS	<i>Advanced Driver-Assistance Systems</i>
AMEC	Acurácia Média dos Experimentos por Contexto
AMECC	Acurácia Média dos Experimentos por Contexto e Colocação
ATIS	<i>Advanced Traveler Information Systems</i>
ATMS	<i>Advanced Traffic Management Systems</i>
APTMS	<i>Advanced Public Transport Management Systems</i>
AVCS	<i>Advanced Vehicle Control Systems</i>
CID	<i>Complexity Invariant Distance</i>
CNN	<i>Convolutional Neural Network</i>
CNN-LSTM	<i>Convolutional Neural Network Long Short-Term Memory</i>
CNT	Confederação Nacional do Transporte
ConvLSTM	<i>Convolutional Long Short-Term Memory</i>
DTW	<i>Dynamic Time Warping</i>
DDDTW	<i>Derivative Dynamic Time Warping</i>
DNN	<i>Deep Neural Networks - DNN</i>
DTD	<i>Derivative Transform Distance</i>
FFT	<i>Fast Fourier Transform</i>
FP	Falsos Positivos
FN	Falsos Negativos
FSR	<i>Full Scale Range</i>
GPBi	<i>Global Pave Box index</i>
GPS	<i>Global Positioning System</i>
GRU	<i>Gated Recurrent Unit</i>
GT	<i>Ground Truth</i>
HC	<i>Half-Car</i>

IA	Inteligência Artificial
INPI	Instituto Nacional da Propriedade Industrial
IIR	<i>Infinite Impulse Response</i>
IRAS	<i>Intelligent Road Assessment System</i>
IRI	<i>International Roughness Index</i>
ITS	<i>Intelligent Transport Systems</i>
KMC	<i>K-Means Clustering</i>
KNN	<i>K-Nearest Neighbors</i>
LPBi	<i>Local Pave Box index</i>
LCSS	<i>Longest Common Subsequence Similarity</i>
LSTM	<i>Long Short-Term Memory</i>
MSL	Mapeamento Sistemático da Literatura
OBD-II	<i>On-Board Diagnostic II</i>
PSR	<i>Present Serviceability Rating</i>
PVS	<i>Passive Vehicular Sensors Dataset</i>
RMS	<i>Root Mean Square</i>
QC	<i>Quarter-Car</i>
RQI	<i>Riding Quality Index</i>
RSCI	<i>Road Surface Condition Index</i>
RSL	Revisão Sistemática da Literatura
SBC	<i>Single-Board Computer</i>
SOM	<i>Self Organizing Maps</i>
STFT	<i>Short-Time Fourier Transform</i>
SVM	<i>Support Vector Machines</i>
VP	Verdadeiros Positivos
VN	Verdadeiros Negativos

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>21</b>
1.1	CONTEXTUALIZAÇÃO DO PROBLEMA	23
1.2	OBJETIVOS	24
<b>1.2.1</b>	<b>Objetivo Geral</b>	<b>24</b>
<b>1.2.2</b>	<b>Objetivos Específicos</b>	<b>24</b>
1.3	JUSTIFICATIVA	25
<b>1.3.1</b>	<b>Cenários de Aplicação</b>	<b>26</b>
1.4	CONTRIBUIÇÕES	27
1.5	ESTRUTURA DO TRABALHO	28
<b>2</b>	<b>METODOLOGIA</b>	<b>29</b>
2.1	LEVANTAMENTO DO ESTADO DA ARTE	29
2.2	COLETA DE DADOS	30
2.3	EXPERIMENTOS E VALIDAÇÃO	31
<b>3</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>33</b>
3.1	SENSORIAMENTO	33
<b>3.1.1</b>	<b>Sensores Inerciais</b>	<b>33</b>
<b>3.1.2</b>	<b>Magnetômetro</b>	<b>33</b>
<b>3.1.3</b>	<b>Global Position System</b>	<b>34</b>
3.2	QUARTER CAR	34
3.3	SÉRIES TEMPORAIS	35
3.4	TÉCNICAS CLÁSSICAS DE MACHINE LEARNING	35
<b>3.4.1</b>	<b>K-Means Clustering</b>	<b>36</b>
<b>3.4.2</b>	<b>K-Nearest Neighbors</b>	<b>36</b>
<b>3.4.3</b>	<b>Support Vector Machines</b>	<b>37</b>
3.5	TÉCNICAS DE DEEP LEARNING	37
<b>3.5.1</b>	<b>Long Short-Term Memory</b>	<b>39</b>
<b>3.5.2</b>	<b>Gated Recurrent Unit</b>	<b>40</b>
<b>3.5.3</b>	<b>Convolutional Neural Network</b>	<b>41</b>
<b>3.5.4</b>	<b>Convolutional Long Short-Term Memory</b>	<b>42</b>
<b>3.5.5</b>	<b>Convolutional Neural Network Long Short-Term Memory</b>	<b>43</b>
3.6	MÉTRICAS DE AVALIAÇÃO	43
<b>4</b>	<b>REVISÃO DA LITERATURA</b>	<b>45</b>
4.1	DEFINIÇÕES DA BUSCA	45
4.2	EXECUÇÃO DA BUSCA	48
4.3	ANÁLISE DO ESTADO DA ARTE	48

<b>4.3.1</b>	<b>Referenciais</b> . . . . .	<b>49</b>
<b>4.3.2</b>	<b>Fatores de Dependência</b> . . . . .	<b>50</b>
<b>4.3.3</b>	<b>Coleta de Dados</b> . . . . .	<b>52</b>
<b>4.3.4</b>	<b>Pré-Processamento</b> . . . . .	<b>53</b>
<b>4.3.5</b>	<b>Processamento</b> . . . . .	<b>56</b>
4.3.5.1	<i>Abordagens Baseadas em Limiares</i> . . . . .	57
4.3.5.2	<i>Abordagens de Alinhamento de Séries Temporais</i> . . . . .	58
4.3.5.3	<i>Abordagens de Funções Lineares</i> . . . . .	59
4.3.5.4	<i>Outras Abordagens</i> . . . . .	59
<b>4.3.6</b>	<b>Padrões de Percepção Veicular</b> . . . . .	<b>60</b>
<b>4.3.7</b>	<b>Áreas de Aplicação</b> . . . . .	<b>60</b>
<b>4.3.8</b>	<b>Discussão e Conclusões</b> . . . . .	<b>62</b>
4.4	<b>ANÁLISE DE PERCEPÇÕES VEICULARES ESPECÍFICAS</b> . . . . .	67
<b>4.4.1</b>	<b>Classificação do Tipo de Superfície de Pista</b> . . . . .	<b>68</b>
<b>4.4.2</b>	<b>Classificação da Qualidade de Superfície de Pista</b> . . . . .	<b>68</b>
<b>4.4.3</b>	<b>Reconhecimento de Lombadas</b> . . . . .	<b>72</b>
<b>5</b>	<b>CONJUNTOS DE DADOS</b> . . . . .	<b>77</b>
5.1	REDE DE SENSORES . . . . .	78
5.2	EXECUÇÃO DA COLETA . . . . .	79
5.3	CLASSES DE DADOS . . . . .	81
<b>6</b>	<b>CLASSIFICAÇÃO DE TIPO DE SUPERFÍCIE DE PISTA 1</b> . . . . .	<b>87</b>
6.1	PRÉ-PROCESSAMENTO . . . . .	87
6.2	PROCESSAMENTO . . . . .	90
6.3	ANÁLISE DE RESULTADOS . . . . .	95
<b>7</b>	<b>CLASSIFICAÇÃO DE TIPO DE SUPERFÍCIE DE PISTA 2</b> . . . . .	<b>103</b>
7.1	PRÉ-PROCESSAMENTO . . . . .	103
7.2	PROCESSAMENTO . . . . .	105
7.3	ANÁLISE DE RESULTADOS . . . . .	107
<b>8</b>	<b>CLASSIFICAÇÃO DE QUALIDADE DE SUPERFÍCIE DE PISTA</b> . . . . .	<b>113</b>
8.1	PRÉ-PROCESSAMENTO . . . . .	113
8.2	PROCESSAMENTO . . . . .	115
8.3	ANÁLISE DE RESULTADOS . . . . .	119
<b>9</b>	<b>RECONHECIMENTO DE LOMBADAS</b> . . . . .	<b>125</b>
9.1	PRÉ-PROCESSAMENTO . . . . .	125
9.2	PROCESSAMENTO . . . . .	128
9.3	ANÁLISE DE RESULTADOS . . . . .	131

<b>10</b>	<b>MATERIAIS RESULTANTES</b> . . . . .	<b>137</b>
<b>11</b>	<b>CONCLUSÕES E DISCUSSÕES</b> . . . . .	<b>139</b>
11.1	TRABALHOS FUTUROS . . . . .	140
	<b>REFERÊNCIAS</b> . . . . .	<b>143</b>



## 1 INTRODUÇÃO

Os sistemas de transporte se estabeleceram ao longo da história como um dos principais condicionantes ao desenvolvimento humano. Em sua utilização, seja para circulação de pessoas ou escoamento de produção, intensifica-se cada vez mais demandas por eficiência, dado seu fator estratégico. Neste contexto, com o avanço das tecnologias computacionais, surgiram os Sistemas de Transporte Inteligentes (*Intelligent Transport Systems - ITS*), os quais constituem soluções tecnológicas desenvolvidas para aprimorar o desempenho e a segurança dos sistemas de transporte (ZHANG et al., 2011; ARAGÓN et al., 2016). Nestas aplicações, o ambiente de tráfego é monitorado através de sensores empregados na infraestrutura de transporte e seus participantes (ZHANG et al., 2011; MATHEW, 2014a; MATHEW, 2014b). Estes sensores produzem dados brutos que são posteriormente processados por modelos computacionais para gerar informações situacionais acerca do modal (ZHANG et al., 2011).

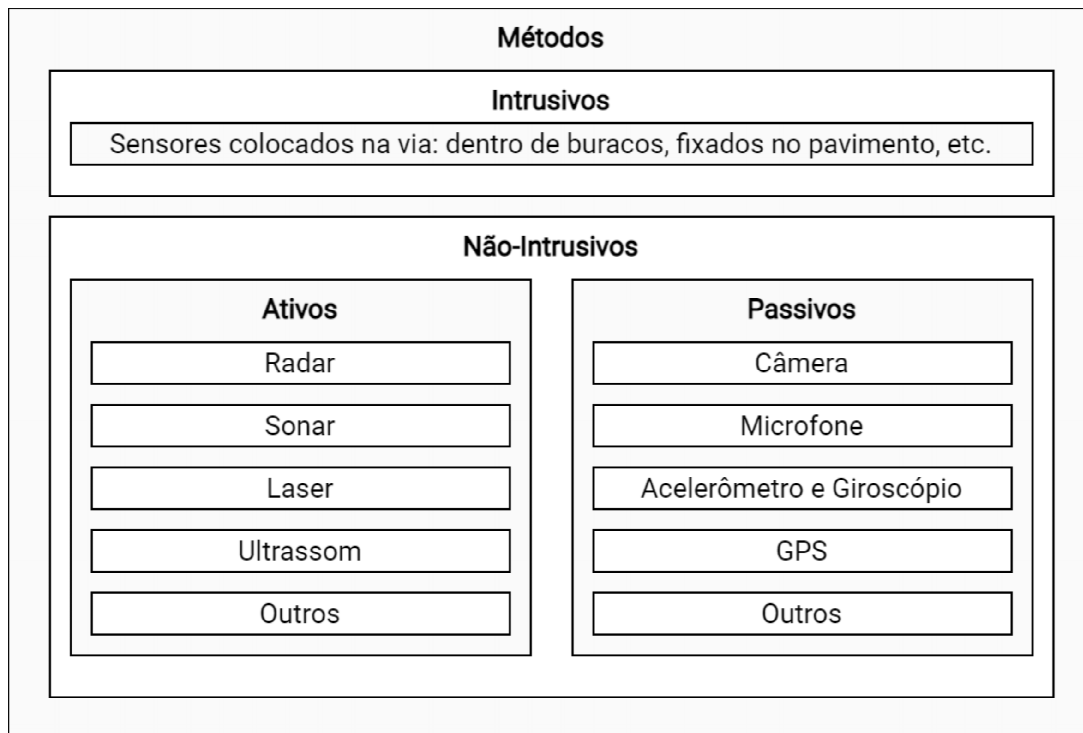
Os dados situacionais produzidos em ITS podem ser classificados de diversas maneiras. Uma delas, a percepção veicular, os sensores são empregados de forma a ser possível produzir exerocepções e propriocepções (MENEGAZZO; WANGENHEIM, 2020). A exerocepção busca compreender o ambiente fora do veículo, realizando o reconhecimento de características do caminho no qual trafega. Estas características incluem eventos transientes, sendo anomalias e obstáculos tais como buracos, trincas em malha, lombadas etc.; e persistentes, como tipo de pavimentação, condição de conservação e qualidade da superfície da pista (faixa de rodagem). A propriocepção, por sua vez, objetiva compreender os movimentos veiculares para identificar seu próprio comportamento. Estas identificações também podem ser transientes, como eventos de condução, do tipo troca de pista, frenagem, derrapagem, aquaplanagem, virando à direita ou esquerda; e persistentes, como perfil de comportamento de condução em seguro ou perigoso (MENEGAZZO; WANGENHEIM, 2018; MENEGAZZO; WANGENHEIM, 2020).

As informações situacionais produzidas possuem grande aplicabilidade, com impactos sociais, econômicos e ambientais. No suporte à tomada de decisão humana, dados como qualidade do pavimento, e geolocalização de obstáculos e irregularidades na pista podem ser empregados em sistemas gerenciais para planejamento de rotas no escoamento de produção, uma vez que o inadequado estado de conservação do pavimento produz, em média, elevação de 27,0% dos custos operacionais (CNT, 2017). Estes dados também podem ser utilizados por administradoras da via para planejamento de manutenções e controle de tráfego, ou por usuários em Sistemas Avançados de Assistência ao Motorista (*Advanced Driver-Assistance Systems - ADAS*), melhorando a segurança no trânsito. No suporte à tomada de decisão artificial, além dos dados supracitados, outros como tipo de pavimento e perfil de condução podem ser utilizados em veículos autônomos para auxiliar na coordenação de suas ações.

Para a produção dos dados brutos que, após processados, geram as informações na forma de percepção veicular, tecnologias com diferentes abordagens foram propostas, dentre métodos intrusivos e não-intrusivos (NI, 2016), conforme ilustra a Figura 1. Na abordagem intrusiva, os dados brutos são amostrados através de sensores colocados diretamente na super-

fície de via, requerendo possíveis alterações no pavimento ou no tráfego (MATHEW, 2014a). Já na abordagem não-intrusiva, não são realizadas alterações na infraestrutura da via, com os sensores colocados dentro dos veículos que nela trafegam (MATHEW, 2014b). Os métodos não-intrusivos possuem diversas vantagens em relação aos intrusivos, tais como serem menos onerosos, de fácil instalação, cobrirem uma área de monitoramento muito maior e permitirem, além da inspeção da infraestrutura, o acompanhamento das ações dos participantes através da propriocepção.

Figura 1 – Abordagens de sensoriamento em ITS



Fonte: Desenvolvido pelo autor.

Na abordagem não-intrusiva, são empregadas técnicas de sensoriamento passivo e ativo. As técnicas ativas requerem interação com o ambiente para produzir seus dados brutos, com a emissão de ondas no ambiente externo através de laser, ultrassom, sonar ou radar. As técnicas passivas, por sua vez, amostram seus dados sem necessitar interagir com o ambiente externo, como dados físicos ou imagens. Em comparação ao sensoriamento ativo, a abordagem passiva é considerada mais segura, não poluente e geralmente de menor custo. Essas características tornam estas técnicas mais interessantes para uso em larga escala, tal como sua aplicação em sistemas tipo do ADAS ou veículos autônomos.

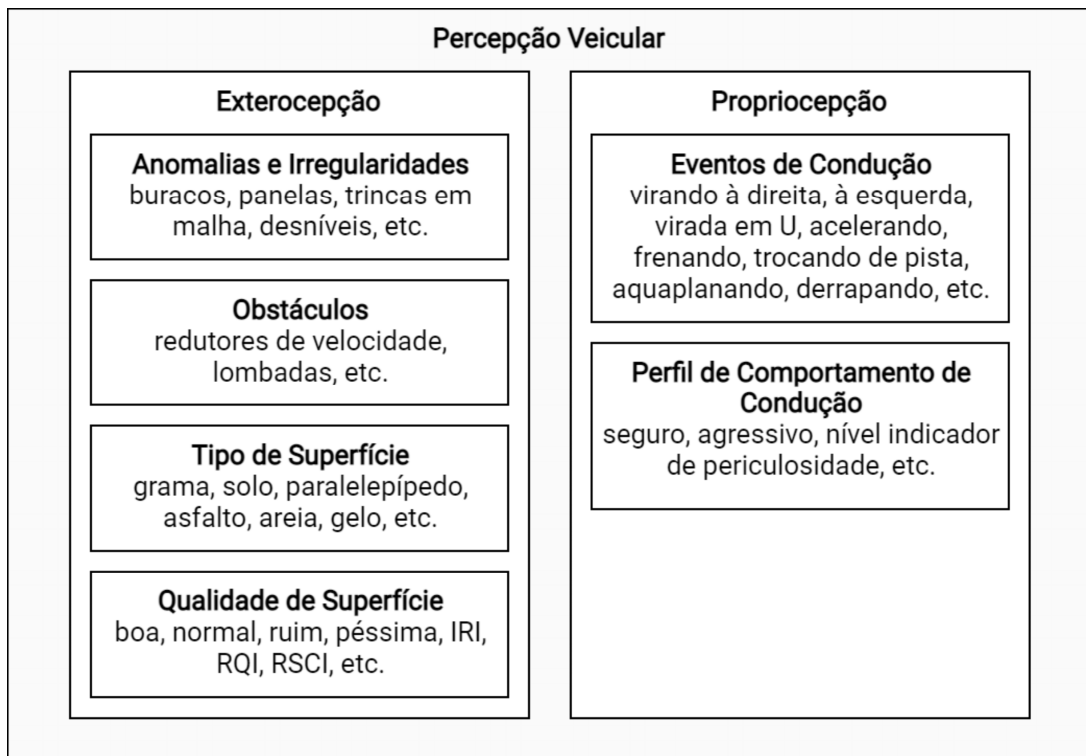
Dentre as soluções de sensoriamento passivo, a utilização de câmeras tem sido amplamente explorada nos últimos anos, com o desenvolvimento de soluções robustas em visão computacional. Contudo, o mesmo não ocorre com os sensores inerciais, representados por acelerômetro e giroscópio, os quais constituem uma alternativa importante a ser melhor explorada (MENEGAZZO; WANGENHEIM, 2018; MENEGAZZO; WANGENHEIM, 2020).



## 1.1 CONTEXTUALIZAÇÃO DO PROBLEMA

Classificados como não-intrusivos e passivos, os sensores inerciais se baseiam no princípio da inércia para produção de seus dados (BRAGA, 2017). Representados por giroscópios e acelerômetros, estes dispositivos produzem, respectivamente, sinais unidimensionais referentes a taxa de rotação e a força de aceleração em seus três eixos físicos (GROVES, 2013). Neste estudo, estes sinais são resultantes da tração do veículo e das interações com o ambiente no qual ele trafega. Através da aplicação em veículos, sejam fixados diretamente na estrutura veicular ou embarcados em dispositivos móveis como *smartphones* e *tablets*, os sensores inerciais permitem a produção de uma grande diversidade de informações situacionais, conforme ilustra a Figura 2.

Figura 2 – Percepções veiculares produzidas através de dados de sensores inerciais



Fonte: Desenvolvido pelo autor.

Embora sejam diversas e relevantes as percepções veiculares produzidas, a utilização de sensores inerciais em ITS pouco avançou nos últimos anos. Através da Revisão Sistemática da Literatura (RSL) (MENEGAZZO; WANGENHEIM, 2018; MENEGAZZO; WANGENHEIM, 2020), foi observado que a maioria dos estudos da área apresenta foco na aplicação das percepções produzidas. Sendo assim, consistem em pesquisas na forma de prova de conceito, onde as soluções apresentadas para produção de percepções veiculares são simplistas e não aplicáveis a cenários do mundo real, devido a inadaptação dos modelos. Contudo, entendemos que a adaptabilidade é um recurso essencial, onde a solução desenvolvida precisa apresentar um certo grau de confiabilidade quando aplicada em cenários não controlados

onde há variações contextuais. Em uma análise comparativa, enquanto soluções baseadas em imagens/vídeos se mostram maduras, com análises em variações contextuais tais como avaliação em diferentes dimensões, marcações desbotadas, diferentes condições de iluminação e oclusão por causa de veículos ou pedestres (SRIMONGKON; CHIRACHARIT, 2017; PATIL; VARIYAR.; SOMAN, 2020), o mesmo não ocorre com os modelos baseados em sinais de sensores inerciais. Nas aplicações desenvolvidas com estes sensores, os modelos analisam cenários controlados e limitados, sem considerar variações de condições no contexto que influenciam os sinais amostrados e, portanto, o resultado da solução final (MENEGAZZO; WANGENHEIM, 2018; MENEGAZZO; WANGENHEIM, 2020).

De acordo com Carlos et al. (2019), nos últimos 10 anos se produziu uma literatura vasta sobre modelos para produzir informações situacionais a partir dos dados de sensores inerciais fixados no veículo ou embarcados nos *smartphones*. Contudo, é necessário em novas pesquisas tornar os modelos mais inteligentes para traçar o perfil das estradas com detalhes reais (CARLOS et al., 2019). Esta afirmação também é sustentada pelos resultados obtidos no levantamento do estado da arte (MENEGAZZO; WANGENHEIM, 2018; MENEGAZZO; WANGENHEIM, 2020), onde observa-se que para um modelo de percepção veicular operar de forma segura e confiável em cenários do mundo real, é necessário considerar os fatores de dependência que impactam nos sinais dos sensores inerciais. Estes fatores estão relacionados à diversidade contextual existente, onde o modelo pode ser aplicado a diferentes veículos, motoristas e ambientes. Considerar estes fatores no modelo o torna adaptável a diferentes condições, sendo este um requisito essencial para sua ampla utilização.

## 1.2 OBJETIVOS

Nesta seção são detalhados o objetivo geral e específicos desta pesquisa.

### 1.2.1 Objetivo Geral

Este trabalho objetiva desenvolver modelos adaptativos para produção de percepções veiculares com emprego de sinais de sensores inerciais e técnicas de Inteligência Artificial (IA). Para satisfazer o requisito de adaptabilidade, os modelos devem demonstrar boa capacidade de generalizar seu aprendizado quando aplicado em contextos desconhecidos, os quais apresentam variações contextuais relacionadas às suas propriedades de dependência.

### 1.2.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Identificar o estado da arte acerca da produção de percepção veicular através de sinais de sensores inerciais;

- Desenvolver uma infraestrutura e uma metodologia para coleta de dados brutos;
- Produzir conjuntos de dados com variações contextuais das propriedades de dependência;
- Desenvolver um modelo para classificação do tipo de superfície de pista, classificando os sinais dos sensores inerciais entre segmentos de terra, paralelepípedo ou asfalto;
- Desenvolver um modelo para classificação da irregularidade de superfície de pista, classificando os sinais dos sensores inerciais entre segmentos de qualidade ruim, regular ou boa;
- Desenvolver um modelo para detecção de lombadas, através de sinais dos sensores inerciais;
- Validar a adaptabilidade dos modelos através de um *design* experimental, onde cada modelo será avaliado quanto a sua capacidade de generalização de aprendizado para contextos desconhecidos;

### 1.3 JUSTIFICATIVA

Para disseminar a utilização de ITS em suas mais variadas formas, mostra-se necessário tornar estes sistemas cada vez mais inteligentes. Para isto, é necessário dispor de mais fontes de dados, que estes dados sejam confiáveis, e que os dispositivos que os produzem não sejam prejudiciais à saúde humana quando aplicados em massa. Sendo assim, os sensores inerciais constituem uma fonte de dados alternativa, com grande potencial de melhorar as aplicações em ITS. Devido a sua abordagem passiva, se mostra um meio seguro, não poluente e de baixo custo. A segurança de sua aplicação em massa já foi amplamente validada, uma vez que todo *smartphone* e *tablet* possui um conjunto destes dispositivos. A poluição, neste contexto, considera um cenário de ampla aplicação, no qual todo carro tem um conjunto de sensores para percepção veicular e, em uma situação de engarrafamento em horário de pico, sensores ativos emitem uma imensa quantidade de ondas no ambiente. Neste mesmo cenário, os sensores passivos não emitem tais ondas, sendo não poluentes.

Os sinais produzidos pelos sensores inerciais permitem a produção de uma grande variedade de percepções veiculares, dentre exterocepções e propriocepções. Contudo, os atuais modelos de percepção veiculares baseados em sensores inerciais não são confiáveis, se mostrando o principal entrave para sua maior adoção. A não confiabilidade nos modelos atuais se deve essencialmente a falta de adaptabilidade das soluções. Desta forma, os modelos desenvolvidos não mantêm sua efetividade quando aplicados em cenários diferentes do qual foi experimentado. Sendo assim, os estudos no estado da arte não consideram as variações contextuais pelas quais o modelo será submetido, e os fatores de dependência relacionados. Desta forma, o desenvolvimento de um modelo adaptativo de percepção se mostra de grande contribuição, seja aplicado na tomada de decisão humana ou artificial. Constituindo um aplicação meio para outros sistemas,

diversas áreas dos ITS podem se beneficiar, como veículos autônomos, Sistemas Avançados de Controle de Veículos (*Advanced Vehicle Control Systems - AVCS*), Sistemas Avançados de Informação ao Viajante (*Advanced Traveler Information Systems - ATIS*), Sistemas Avançados de Gerenciamento de Tráfego (*Advanced Traffic Management Systems - ATMS*), Sistemas Avançados de Gerenciamento de Transporte Público (*Advanced Public Transport Management Systems - APTMS*), ADAS, entre outros (ZHANG et al., 2011; SINGH; GUPTA, 2015).

### 1.3.1 Cenários de Aplicação

Nesta seção são detalhados alguns cenários de aplicação dos modelos propostos. As aplicações podem produzir e utilizar as percepções veiculares tanto localmente quanto remotamente, em servidor central.

**Veículos Autônomo:** Um veículo equipado com sensores inerciais e auxiliares de suporte trafega em uma via. A irregularidade longitudinal da pista, correspondente ao conjunto dos desvios da superfície, é processada em relação a um plano de referência. Nesta análise, é estabelecido um índice de qualidade de conservação e identificado o tipo de pavimentação da via. Também são reconhecidos e classificados eventos transientes de percepção de ambiente, como obstáculos na via (lombadas, redutores de velocidade etc.), deficiências da superfície (buracos, solavancos etc.); e de propriocepção, como eventos de condução (virando à direita suave ou bruscamente, virando à esquerda, acelerando, freando etc.). Disponibilizando estas informações através de uma interface ao agente inteligente que controla o veículo, o índice de qualidade de superfície e o tipo de pavimentação aferido são empregados no controle de velocidade veicular, sendo menor em vias mais irregulares e vice-versa. Esta decisão pode ser monitorada através dos eventos de condução e, se necessário, efetuado ajustes de comando. Os eventos transientes detectados de percepção de ambiente, utilizados na forma de evidências, auxiliam na convalidação de dados obtidos de sensores multimodais, tal qual por intermédio de visão computacional, corroborando hipóteses sobre o contexto no qual está inserido.

**Sistema Avançado de Assistência ao Motorista:** Em um sistema de *vehicular crowdsensing* com sensoriamento oportunista, um *crowdsourcer* utiliza um aplicativo de ADAS em seu computador de bordo ou *smartphone*. Com os sensores inerciais anexados ao veículo ou embarcados nos dispositivos móveis, o aplicativo analisa as vibrações recebidas, para estabelecer conceitos qualitativos sobre a irregularidade de superfície e reconhecer eventos transientes, na forma de obstáculos e anomalias. Um servidor central recebe dados de vários *crowdsourcers*, onde aplica-se um aprendizado baseado em reforço. Empregando a confiabilidade inicialmente estabelecida e a quantidade de detecções em diferentes fontes, a aplicação central realiza ajustes periódicos dos valores de confiabilidade dos dados, de forma que falsos positivos tenham progressivamente sua confiabilidade reduzida a zero, assim como algum evento transiente que deixe de existir, especialmente em função das

manutenções realizadas na via. Com esse processamento no servidor, os dados atualizados são baixados pelos *crowdsourcers*, de forma que o aplicativo ADAS utiliza-os para auxiliar o motorista, alertando-o sobre a velocidade acima da segura para uma pista com aquela qualidade de superfície, obstáculos e deficiências durante o trajeto.

**Sistema de Monitoramento de Infraestrutura de Transporte Terrestre:** Um veículo equipado com sensores inerciais e auxiliares de suporte trafega em uma via. A irregularidade longitudinal da pista, correspondente ao conjunto dos desvios da superfície, é processada em relação a um plano de referência. Nesta análise, é estabelecido um índice de qualidade de conservação e identificado o tipo de pavimentação da via. Também são reconhecidos e classificados eventos transientes de percepção de ambiente, como obstáculos na via (lombadas, redutores de velocidade etc.), deficiências da superfície (buracos, solavancos etc.). Estas informações situacionais são salvas em um servidor remoto, com as respectivas coordenadas geodésicas. Posteriormente, estes dados são utilizados para definir a manutenção das vias, quando e onde deve ocorrer. Estes dados podem integrar relatórios governamentais, tais como o Relatório Gerencial produzido pela Confederação Nacional do Transporte (CNT), de forma a auxiliar direcionamento de investimentos públicos no modal de transporte terrestre.

#### 1.4 CONTRIBUIÇÕES

Este trabalho apresenta diversas contribuições de aspecto teórico e aplicado. Com a produção de uma ampla revisão sistemática da literatura, foi realizado o levantamento do estado da arte na aplicação de sensores inerciais para produção de percepções veiculares. Através da análise deste levantamento, foram mapeados todos os fatores de dependência existentes no contexto de ITS que influenciam os sinais amostrados através de sensores inerciais. Este mapeamento, até então inexistente, é importante para que se possa desenvolver modelos de percepção veicular mais robustos, com adaptabilidade ao contexto. Junto a estes fatores, foram produzidos mapeamentos de aspectos diversos, que compreendem desde a etapa da coleta de dados, pré-processamento e processamento.

Com o estabelecimento do estado da arte e mapeamento de aspectos importantes nesta área de pesquisa, foi desenvolvida uma metodologia de coleta de dados, a qual compreende desde a criação da rede de sensores, referenciais de coleta e análise, colocação e posicionamento dos sensores na infraestrutura veicular, etc. Através dessa metodologia, foram produzidos nove conjuntos de dados com variações contextuais relacionadas aos fatores de dependência: diferentes veículos, motoristas e ambientes. Baseando-se nos conjuntos de dados, produzimos um *design* experimental compreendendo as etapas de treinamento e validação dos modelos, de forma a ser possível avaliar a eficiência da generalização de aprendizado do modelo quando aplicado a um contexto desconhecido. Por fim, através de técnicas de *Deep Learning* foram desenvolvidos três modelos adaptativos de percepção veicular, de forma que suas camadas de

processamento fossem capazes de compreender as relações entre os dados e seus fatores de dependência. O primeiro modelo classifica o tipo de superfície de pista. O segundo, a qualidade da superfície. Por fim, o terceiro identifica lombadas na via.

Este projeto é inteiramente *open-source*, disponibilizado no Github <sup>1</sup> <sup>2</sup>. Desta forma, os códigos-fonte utilizados na coleta de dados, para manipulação de sensores, amostragem e armazenamento dos sinais; no pré-processamento, com ajustes, combinação de dados, normalização, etc.; no processamento, com a classificação dos padrões; assim como os próprios datasets e modelos desenvolvidos, estão documentados e disponíveis publicamente, permitindo pesquisas futuras executarem, compararem e auditarem os experimentos. Convém mencionar que os conjuntos de dados produzidos são provavelmente os primeiros do tipo a serem disponibilizados publicamente, com coleta em múltiplos pontos e emprego de diversos sensores de abordagem passiva. Adicionalmente, o projeto conta com diversos materiais de divulgação científica, como vídeos dos sinais e modelos operando, e *Jupyter Notebooks* com fundamentação teórica junto do código-fonte dos melhores modelos. Desta forma, estes materiais auxiliam na popularização deste tipo de sensoriamento em ITS, especialmente por atenuar a curva de aprendizado de novos pesquisadores na área.

## 1.5 ESTRUTURA DO TRABALHO

Esta dissertação está estruturada em onze capítulos. No Capítulo 1, é introduzido o problema de pesquisa, contextualização, justificativa, objetivos, contribuições sociais e científicas para a computação. No Capítulo 2 é apresentada a metodologia utilizada neste trabalho, dentre os métodos para levantamento do estado da arte, coleta de dados, desenvolvimento e validação dos modelos. No Capítulo 3 é discorrida a fundamentação teórica necessária para compreensão desta pesquisa. No Capítulo 4 é detalhada a revisão da literatura produzida e resultados obtidos, especificando as lacunas de pesquisa existentes com enfoque nas três percepções trabalhadas. No Capítulo 5 é detalhada a metodologia de coleta de dados, rede de sensores produzida, referenciais adotados, colocação, posicionamento e configuração, além dos nove conjuntos produzidos. Nos Capítulos 6 a 9 é apresentado a metodologia de desenvolvimento e validação, e os resultados obtidos do modelo adaptativo para classificação de superfície, classificação de qualidade e detecção de lombadas, respectivamente. Por fim, no Capítulo 10 são apresentados os materiais resultantes e no Capítulo 11 estão as considerações finais e sugestões de trabalhos futuros.

---

<sup>1</sup> <https://github.com/Intelligent-Vehicle-Perception/Intelligent-Vehicle-Perception-Based-on-Inertial-Sensing-and-Artificial-Intelligence>

<sup>2</sup> <https://codigos.ufsc.br/lapix/intelligent-vehicle-perception-based-on-inertial-sensing>

## 2 METODOLOGIA

Este trabalho utiliza do método científico hipotético-dedutivo proposto pelo filósofo austríaco Karl Popper (POPPER, 2002). Neste método, partindo de hipóteses estabelecidas, são eleitas para experimentação aquelas que possuem certa viabilidade para responder um determinado problema de natureza científica. Após essa eleição, busca-se o falseamento das hipóteses, objetivando comprovar sua sustentabilidade. O método termina com a comprovação das hipóteses; ou com a construção de novas, caso as atuais sejam refutadas (BONAT, 2009).

A metodologia deste trabalho se baseia nas quatro práticas operacionais que consagram o método científico: a identificação de um problema, a indicação de uma hipótese, a coleta de dados e a análise da resposta (BONAT, 2009). Desta forma, partindo da definição do problema de pesquisa, foi realizado o levantamento do estado da arte através de uma revisão sistemática da literatura. Com este levantamento, o problema foi refinado para criação da seguinte hipótese:

**Hipótese:** Através de um conjunto de dados que bem represente a diversidade contextual envolvida na aplicação dos sensores inerciais em ITS para geração de percepção veicular, ou seja, que os dados contenham variações contextuais significativas em relação aos fatores de dependência da solução, é possível construir modelos de Inteligência Artificial capazes de aprender as relações e as influências dos fatores de dependência nos sinais dos sensores, possibilitando a generalização de seu aprendizado para cenários desconhecidos de forma confiável.

Partindo desta hipótese, foi realizada a coleta de dados respeitando a metodologia necessária para efetuar o estudo de validação. Para este estudo, foi definido um *design* experimental no qual se pudesse testar a veracidade da hipótese levantada. Por fim, de forma a mensurar os resultados dos experimentos para comprovar ou refutar a hipótese, métricas específicas de avaliação dos modelos foram adotadas. Cada uma destas etapas é detalhada nas próximas seções.

### 2.1 LEVANTAMENTO DO ESTADO DA ARTE

Esta etapa tem como objetivo o levantamento do estado da arte acerca da aplicação de sensores inerciais em ITS para geração de percepção veicular. Desta forma, foi conduzida uma Revisão Sistemática da Literatura (RSL) baseada nos procedimentos descritos em (KITCHENHAM, 2004; KITCHENHAM et al., 2009; BIOLCHINI et al., 2005). A RSL foi dividida em três fases: definição, execução e análise. Na primeira, foi delimitado o escopo de busca, definidas as perguntas de pesquisa, especificadas as bases científicas utilizadas e suas *strings* de busca, além dos critérios de inclusão e exclusão dos estudos primários. Na segunda fase, foi executada a busca e selecionados os trabalhos que aderiram aos critérios de seleção definidos na fase anterior. Por fim, na terceira fase, cada estudo foi analisado na íntegra, com extração e sumarização das informações de interesse.

A revisão produzida buscou responder as perguntas de pesquisa, mapeando aspectos importantes das etapas de coleta de dados, pré-processamento e processamento. Desta forma, relacionado a coleta de dados, identificou-se o contexto no qual a amostragem de sinais ocorreu, as plataformas de *hardware* utilizadas, os sensores empregados e suas configurações tais como taxa de amostragem e quadros referenciais, além da forma de colocação e posicionamento dos dispositivos na infraestrutura veicular. Em relação à etapa de pré-processamento, foram identificados os métodos e técnicas aplicadas para organizar e transformar os dados antes de serem utilizados como entrada dos modelos de IA, tais como reorientação de eixos, filtragem e normalização de sinais, extração de características, segmentação de dados, etc. Por fim, na etapa de processamento foram identificadas as técnicas utilizadas para reconhecimento dos padrões. Através de uma análise ampla sobre as três etapas percorridas, foram levantados também os fatores de dependência que impactam na adaptabilidade da solução, os tipos de percepção possíveis de serem produzidos, além das áreas de aplicação.

## 2.2 COLETA DE DADOS

Através de nossa RSL, não encontramos nenhum conjunto de dados público de sensores inerciais embarcados em carros que permitisse a análise proposta. Sendo assim, produzimos nove conjunto de dados através de diversos sensores de abordagem passiva. Foram utilizadas duas redes de sensores, sendo cada uma delas constituídas por um *Single-Board Computer* (SBC) Raspberry Pi e três placas MPU-9250, cada uma equipada com um acelerômetro, um giroscópio, um magnetômetro e um sensor de temperatura. Também foi empregada uma fonte de *Global Positioning System* (GPS), com produção de dados de localização e velocidade, além de uma câmera para captura de vídeo ambiente.

A coleta de dados foi realizada de forma a amostrar dados em variações contextuais dos fatores de dependência. Estes fatores são considerados *propriedades sensoriais*, como faixa de medição, resolução, referenciais de captação e análise; *propriedades veiculares*, como sistema de suspensão e mecânica veicular; *propriedades de condução*, como velocidade aplicada ao veículo e estilo de condução; e *propriedades ambientais*, como alterações de condição de conservação, presença de irregularidades e obstáculos, etc. Sendo assim, os sensores foram distribuídos no veículo através de posicionamento controlado, onde a colocação dos módulos foi realizada de forma que os três eixos do sistema de coordenadas do sensor ficaram alinhados com os do veículo, sendo tanto referencial de coleta como de análise. Também foram definidas configurações adequadas para não saturar as leituras dos sensores, tais como *Full Scale Range* (FSR) e resolução, além da taxa de amostragem, satisfazendo a dependência da primeira propriedade.

Para se adequar a metodologia de validação proposta, as três propriedades restantes necessitam apresentar variações contextuais. Sendo assim, em adição a amostragem de velocidade (propriedade de condução) e colocação dos sensores em três diferentes pontos da infraestrutura veicular (propriedade veicular), nós realizamos as coletas de dados com o conjunto de senso-



res aplicados a três diferentes veículos (propriedade veicular), com três diferentes motoristas variando a velocidade de 0 *km/h* até 91,98 *km/h* (propriedade de condução), cobrindo três cenários distintos (propriedade ambiental). Cada cenário conta com três tipos de superfícies, dentre pavimentadas (paralelepípedo e asfalto) e não pavimentadas (terra), além de variações ambientais gerais, tais como presença de lombadas, buracos, variação na conservação dos pavimentos, aclives e declives, etc.

Com os dados coletados, foram criadas as classes de dados a serem utilizadas como *Ground Truth* (GT) dos experimentos. Para a classificação de tipo de superfície de pista, foram criadas as classes terra, asfalto e paralelepípedo. Para a classificação de qualidade de pista, criou-se os níveis ruim, regular e bom. Para a detecção de lombadas, criou-se as classes com e sem lombada. As classes de tipo de superfície e presença de lombadas remetem a elementos objetivos, onde visualmente pode-se estabelecer seu início e fim. Logo, estas duas contam com GT de anotação humana (KRIG, 2014). Já a qualidade de superfície torna-se subjetiva se realizada com este tipo de GT. Sendo assim, desenvolvemos um método de extração de características de vibração que envolvem as três últimas propriedades de dependência. Logo, as classes referentes aos níveis de qualidade contam com GT de anotação automatizada por máquina (KRIG, 2014).

### 2.3 EXPERIMENTOS E VALIDAÇÃO

Os modelos de IA desenvolvidos foram experimentados de forma a verificar a técnica mais adequada para o reconhecimento dos padrões, dentre abordagens de *Machine Learning* clássico e *Deep Learning*, além de avaliar aspectos diversos, como janela de dados ideal, domínio de análise, ponto de coleta, etc. Contudo, para validar a hipótese deste estudo, nós desenvolvemos um *design* experimental com variação das propriedades de dependência veiculares, de condução e ambientais. Convém mencionar que a propriedade sensorial é a única que não depende do aprendizado do modelo, mas sim da metodologia de coleta de dados. Sendo assim, enquanto as demais influenciam o valor quantificado nas medições, podendo aumentar ou diminuir a amplitude, as propriedades sensoriais influenciam a corretude do valor, onde suas configurações e posicionamento podem resultar na saturação dos dados ou referencial incorreto.

Os nove conjuntos produzidos se relacionam proporcionalmente com as três propriedades de dependência de interesse nessa etapa, através da combinação de três veículos, três motoristas e três cenários/ambientes. Sendo assim, a metodologia de validação consiste em três experimentos, onde em cada um deles existe duas variações de propriedade para treinamento, e a terceira variação aparece somente em validação. Desta forma, analisamos o a generalização de aprendizado que o modelo obteve através de dois veículos, quando submetido a um terceiro. O mesmo ocorre com motoristas e ambientes. Sendo assim, conseguimos avaliar o comportamento do modelo quando submetido a um carro, motorista ou ambiente desconhecido. Para mensurar os resultados, utilizamos das métricas de acurácia, precisão, *recall* e *f1-score*.



### 3 FUNDAMENTAÇÃO TEÓRICA

Nesta seção são detalhados fundamentos teóricos necessários para compreensão da pesquisa. Inicialmente é discorrido sobre o sensoriamento empregado, tipos de sensores, valores aferidos, entre outras características. Posteriormente, é apresentado o modelo matemático *Quarter Car* (QC), o qual descreve as mecânicas veiculares envolvidas no sistema de suspensão. Em seguida, são detalhadas as técnicas de IA utilizadas com o intuito de reconhecer e classificar os padrões de percepção veicular, dentre técnicas *Machine Learning* clássico e *Deep Learning*. Por fim, são apresentadas as métricas de avaliação adotadas.

#### 3.1 SENSORIAMENTO

Nesta seção, são apresentados os sensores inerciais, os quais constituem fonte principal de dados do estudo. Em seguida, é discorrido sobre o sensor magnetômetro e GPS, uma vez que servem como sensores auxiliares, comumente utilizados em conjunto com os inerciais para prover informações adicionais.

##### 3.1.1 Sensores Inerciais

Os sensores inerciais constituem dispositivos que produzem sinais através do princípio da inércia (BRAGA, 2017). Estes sensores compreendem acelerômetros e giroscópios, de um ou mais eixos (BEEBY, 2004). O acelerômetro mede a força de aceleração (em  $m/s^2$  ou  $g$ ), enquanto o giroscópio afere a taxa de rotação (em graus/segundo ou radianos/segundo), ambos sem a necessidade de um referencial externo (GROVES, 2013). Sendo assim, a partir de um quadro de referência ou posição inicial destes sensores, forças externas que atuam sobre eles causam acelerações e mudanças de orientação (rotação) em um ou mais de seus eixos (KEMPE, 2011). Em síntese, são sensores que necessitam de movimento para produção de dados. Neste estudo, estes movimentos são resultantes das forças produzidas pela tração do veículo e pelas interações com o ambiente no qual ele trafega.

##### 3.1.2 Magnetômetro

O magnetômetro é um sensor auxiliar comumente embarcado com os sensores inerciais em um mesmo módulo. Este sensor, também de abordagem passiva, mede o campo geomagnético ambiental (em  $\mu T$ ) em seus três eixos físicos (SATTAR; LI; CHAPMAN, 2018). Sendo assim, é geralmente aplicado junto aos ângulos de Euler para dar orientação aos dados, reorientando para o referencial da terra. Também é empregado em conjunto com os dados de aceleração para estimar a localização e velocidade mais rapidamente, uma vez que a taxa de amostragem do GPS é muito mais lenta que a dos sensores inerciais.

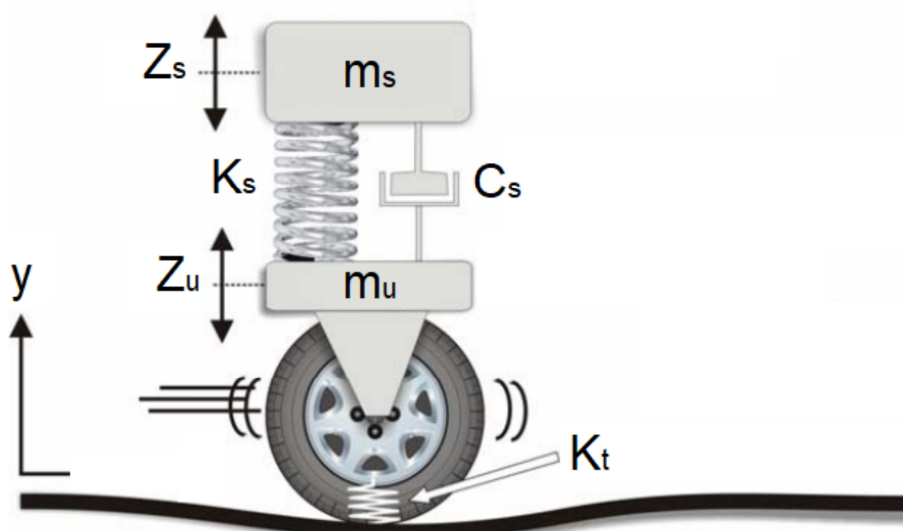
### 3.1.3 Global Position System

O Sistema de Posicionamento Global (*Global Positioning System* - GPS) consiste de um sistema de satélites que orbitam o planeta, auxiliando receptores em terra a determinar sua localização (MILETTE; STROUD, 2012). Sendo assim, além dos dados geodésicos de latitude e longitude, o receptor GPS também afere a velocidade do objeto que contém o sensor. Embora preciso, este sistema possui uma taxa de amostragem baixa em relação aos demais sensores, cerca de 1Hz.

### 3.2 QUARTER CAR

O modelo matemático *Quarter Car* (QC) descreve as variáveis da dinâmica veicular, conforme ilustra a Figura 3. O modelo QC possui propriedades relacionadas a roda, suspensão e amortecimento. A propriedade de massa suspensa do veículo  $m_s$ , fica acima da suspensão e representa um quarto da massa veicular. A propriedade de massa não suspensa  $m_u$ , abaixo da suspensão, inclui a massa de uma roda e do sistema de suspensão conectado a ela. Entre as massas, existe uma suspensão feita de mola com uma taxa de suspensão  $K_s$ , e de um amortecedor com uma taxa de amortecimento  $C_s$ , os quais suportam a massa suspensa. Uma vez que a massa não suspensa está em contato direto com a superfície da pista, existe a rigidez do pneu e sua capacidade de absorção representadas pela taxa de amortecimento do pneu  $K_t$  (AL-YAFEAI; DARABSEH; MOURAD, 2019; LOIZOS; PLATI, 2008). Logo, uma força que parte da superfície da pista atingindo o pneu, será irradiada para todos estes componentes, onde será suavizada pelo deslocamento de massa suspensa  $Z_s$  e da massa não suspensa  $Z_u$ , antes de chegar a parte superior do veículo.

Figura 3 – O modelo *Quarter Car*



Os valores das variáveis do modelo QC são definidos de acordo com cada veículo, dadas as diferentes estruturas veiculares existentes. Este modelo conta com diversas fórmulas que descrevem o relacionamento entre todas as suas variáveis. Os parâmetros do modelo são essenciais na aplicação de metodologias internacionais para indexação da irregularidade, como o *International Roughness Index* (IRI), para estabelecer a qualidade da via. O detalhamento do método para cálculo IRI utilizando o QC pode ser encontrado em Du et al. (2014). De forma a simplificar as fórmulas do QC, foram criados valores padrões para as propriedades, denominados *Golden Car Parameters*. Estas propriedades, detalhadas na Tabela 1, foram estabelecidas considerando um cenário padrão, onde o veículo é dirigido a uma velocidade constante de 80 km/h (LOIZOS; PLATI, 2008).

Tabela 1 – Os parâmetros *Golden Car*

<b>Parâmetro</b>	<b>Valor</b>
$K_s/m_s$	63,3
$K_t/m_s$	653
$C_s/m_s$	6
$m_u/m_s$	0,15

Fonte: Adaptado de (LOIZOS; PLATI, 2008)

### 3.3 SÉRIES TEMPORAIS

As séries temporais consistem de conjuntos de observações ordenadas ao longo do tempo. Alguns exemplos são valores mensais de temperatura, cotação diária do *Bitcoin* e precipitação atmosférica anual. Estas séries podem ser contínuas, sendo aquelas constituídas de observações feitas continuamente no tempo, ou discretas, quando as observações são amostradas da série temporal contínua em intervalos de tempos iguais, em uma frequência de amostragem. Com o objetivo de construir modelos para determinados propósitos através das séries temporais, existem as análises realizadas no domínio do tempo e outras no domínio da frequência (MORETTIN; TOLOI, 2018).

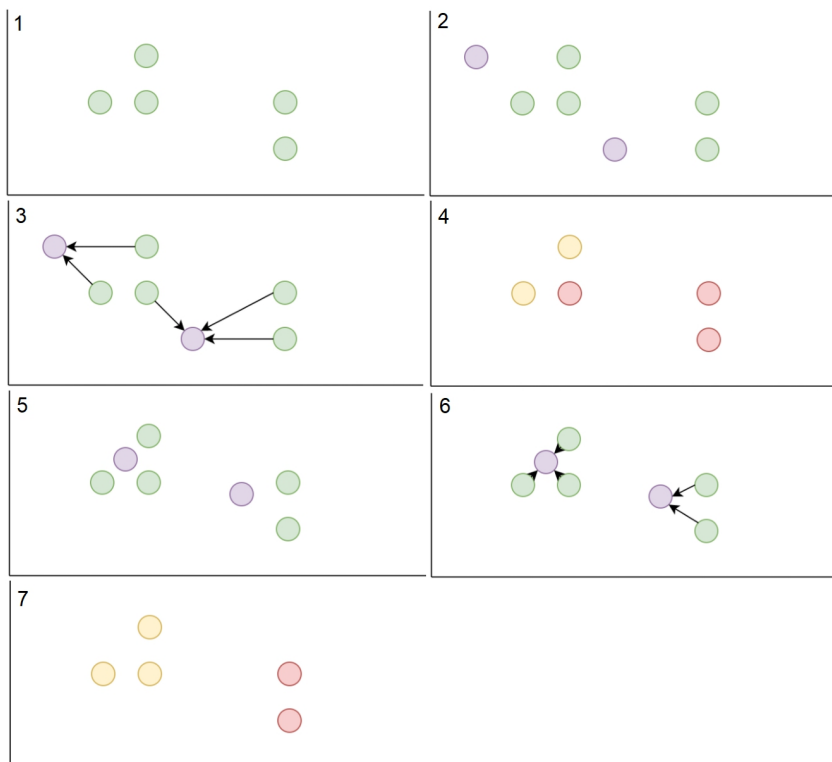
### 3.4 TÉCNICAS CLÁSSICAS DE MACHINE LEARNING

Com o desenvolvimento da Inteligência Artificial, diversas técnicas surgiram para resolver problemas de classificação e regressão (previsão). Dentre as técnicas clássicas de *clustering* e aprendizado de máquina supervisionado, neste trabalho foram desenvolvidos modelos de *K-Means Clustering* (KMC), *K-Nearest Neighbors* (KNN) e *Support Vector Machines* (SVM), detalhadas nas próximas subseções.

### 3.4.1 K-Means Clustering

O KMC consiste de uma técnica não supervisionada para agrupamento de dados (*clustering*), que permite identificar agrupamentos (*clusters*) de dados semelhantes, conforme ilustra a Figura 4. Neste método, partindo de um conjunto de dados não rotulados (1), são criadas aleatoriamente centróides para cada um dos  $k$  *clusters* (2). Em seguida, cada dado é atribuído a um dos *clusters* baseando-se em uma métrica de distância, como a Euclidiana, em relação ao centróide do *cluster* (3, 4). Iterativamente, o centróide de cada *clusters* é recalculado com base na média dos dados (5), e as distâncias novamente recalculadas (6, 7), até que as centroides sejam estabilizadas, ou o número máximo de iterações atingido (FOLEY, 2019; NISBET; ELDER; MINER, 2009). A lista completa de hiperparâmetros e seus valores padrões são encontradas em Scikit-Learn (2021a).

Figura 4 – Execução da técnica KMC

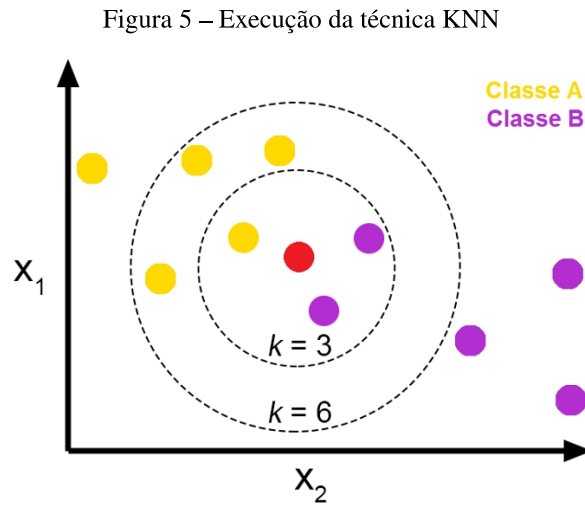


Fonte: Adaptado de (CHRIS, 2020)

### 3.4.2 K-Nearest Neighbors

O KNN consiste de uma técnica para classificação a qual se baseia em métricas de similaridades entre os dados para reconhecer padrões. Desta forma, para um novo dado a ser classificado é calculada a distância entre ele e cada um dos dados de treinamento, identificando os  $k$  vizinhos mais próximos. A classe do novo dado é definida como aquela mais comum entre

seus  $k$  vizinhos, conforme ilustra a Figura 5 (KHANDELWAL, 2018). A lista completa de hiperparâmetros e seus valores padrões são encontradas em Scikit-Learn (2021b).



Fonte: (JOSé, 2018)

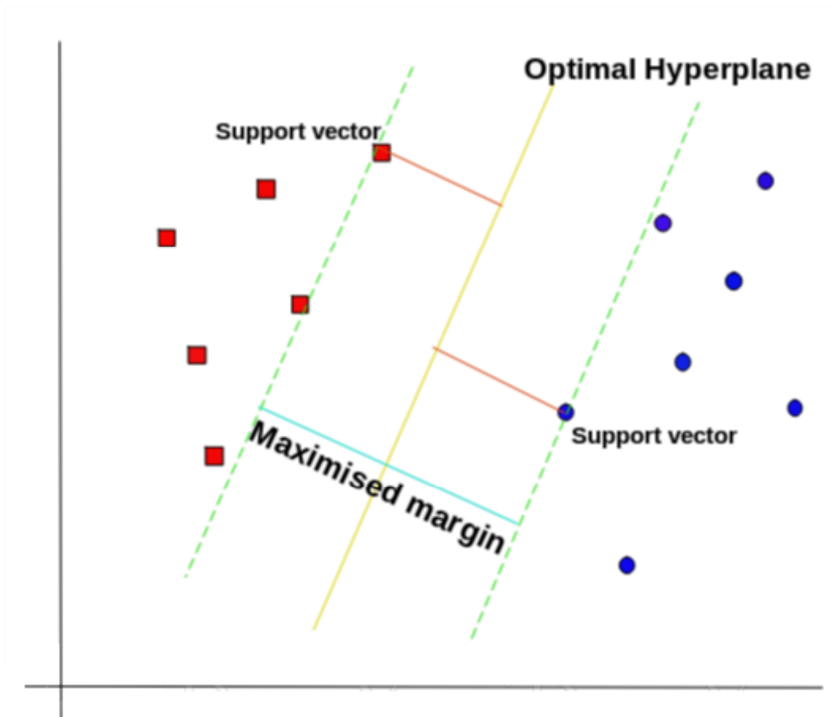
### 3.4.3 Support Vector Machines

O SVM é uma técnica de aprendizado supervisionado aplicada em classificação, onde se busca por um hiperplano ideal que separa as classes de dados, conforme ilustra a Figura 6. O algoritmo se inicia com a busca pelos pontos mais próximos ao hiperplano que separa as classes. Esses pontos são chamados de vetores de suporte. Na sequência, é calculada a distância entre o hiperplano e estes vetores. Essa distância é chamada de margem, a qual o SVM busca maximizar, de forma a encontrar o hiperplano ideal (PUPALE, 2019). Em problemas não lineares, como é o caso deste, é necessário a utilização do hiperparâmetro *kernel*, o qual converte problema não separável em problema separável, para que o SVM possa classificar (SHUBHAM, 2018). O suporte multiclasse, neste estudo, é tratado de acordo com um esquema *one-vs-one*. A lista completa de hiperparâmetros e seus valores padrões são encontradas em Scikit-Learn (2021c).

## 3.5 TÉCNICAS DE DEEP LEARNING

Com o desenvolvimento do aprendizado de máquina, especialmente a partir de 2006, surgiu nesta área um segmento denominado aprendizado profundo (*Deep Learning*) (DENG; YU et al., 2014). Até então, a extração características importantes que bem representassem os dados parametrizados à uma técnica de reconhecimento de padrões, constituía um problema central. Sendo assim, as técnicas existentes apresentavam a grande dificuldade de se extrair, em um pré-processamento, características de alto nível através de dados brutos (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 6 – Execução da técnica SVM



Fonte: (PUPALE, 2019)

Com o desenvolvimento do *Deep Learning*, esse problema foi solucionado, sendo introduzidas representações que são expressas em termos de outras representações, construindo conceitos complexos através de conceitos simples (GOODFELLOW; BENGIO; COURVILLE, 2016). Nessa abordagem, tornou-se possível construir modelos computacionais compostos de múltiplas camadas de processamento para aprender a representação dos dados com diversos níveis de abstração (LECUN; BENGIO; HINTON, 2015). Sendo assim, o aprendizado profundo descobre uma estrutura complexa em grandes conjuntos de dados utilizando do algoritmo de retropropagação para calcular como alterar seus parâmetros internos, os quais são usados para estabelecer a representação em cada camada a partir da representação obtida na camada anterior (LECUN; BENGIO; HINTON, 2015).

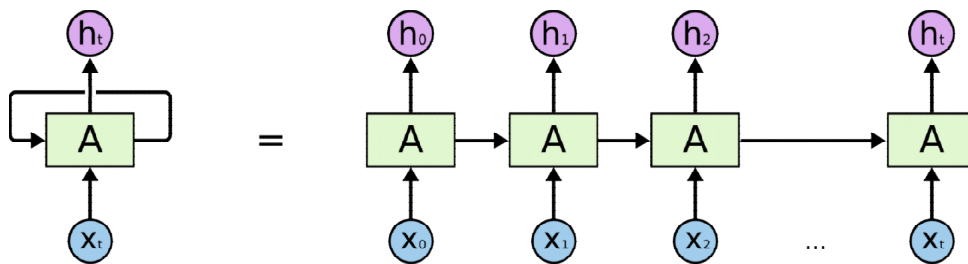
Com este novo paradigma, foram melhorados drasticamente o estado da arte em reconhecimento de fala, reconhecimento visual de objetos, detecção de objetos e muitos outros domínios (LECUN; BENGIO; HINTON, 2015). Os diversos métodos presentes nessa categoria podem ser classificados entre redes profundas para aprendizado não supervisionado ou generativo, redes profundas para aprendizado supervisionado e redes profundas híbridas (DENG; YU et al., 2014). Neste trabalho, foram utilizadas redes profundas para aprendizado supervisionado do tipo *Long Short-Term Memory* (LSTM), *Convolutional Long Short-Term Memory* (ConvLSTM), *Gated Recurrent Unit* (GRU) e *Convolutional Neural Network* (CNN), assim como redes profundas híbridas do tipo *Convolutional Neural Network Long Short-Term Memory* (CNN-LSTM), discorridas nas próximas seções.



### 3.5.1 Long Short-Term Memory

A LSTM constitui um tipo de Rede Neural Recorrente (*Recurrent Neural Network - RNN*) considerada ideal para predição e classificação de séries temporais, substituindo muitas abordagens tradicionais por *Deep Learning* (ZACCONE; KARIM; MENSRAWY, 2017). Suas aplicações envolvem previsão ou classificação de séries temporais (ZACCONE; KARIM; MENSRAWY, 2017), tais como previsão de velocidade de tráfego, modelagem de linguagem, reconhecimento de fala, aprendizagem de gramática, modelagem de áudio, reconhecimento de caligrafia, etc. (BIANCHI et al., 2017). Nesta abordagem, introduzindo o conceito de estado de célula (memória) e *auto-loops* conforme ilustra a Figura 7, a rede pode manter valores por um curto ou longo período de tempo, como uma função de suas entradas. Assim, a célula consegue se lembrar daquilo que é importante, e não apenas do último valor computado (JONES, 2017), de forma que informações relevantes de entradas anteriores são retidas e utilizadas para alterar a saída atual (ZEBIN et al., 2018).

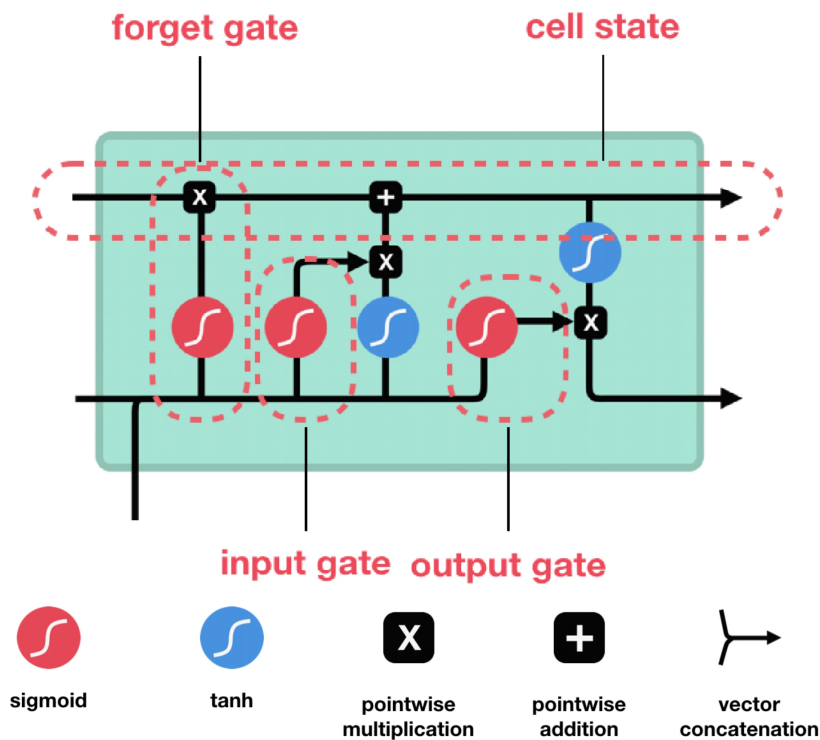
Figura 7 – Célula de uma LSTM com *auto-loops*



Fonte: (JUNIOR, 2019)

Cada célula de uma LSTM possui três portas que controlam o fluxo de informações para dentro e fora da célula, ilustrado na Figura 8. Estas são: a porta de entrada (*input gate*), a porta de saída (*output gate*) e a porta de esquecimento (*forget gate*). As portas possuem pesos, onde os valores computados seguem um fluxo no qual resultam em alterações no estado de célula. Sendo assim, inicialmente tem-se a porta de esquecimento, responsável por decidir quais dados devem ser mantidos na memória e quais devem ser esquecidos (PHI, 2020). Isso permite que a célula se lembre de dados anteriores importantes, ou apenas de novos dados, esquecendo os anteriores. Em seguida, a porta de entrada é responsável por controlar quando novas informações podem entrar na memória. Finalmente, a porta de saída controla quais as informações contidas no próximo estado da célula (JONES, 2017). Cada porta conta com funções de ativação, sendo *tanh* utilizada para regular os valores dos vetores da rede no intervalo  $[-1,1]$ , e a *sigmoid* para regular entre  $[0,1]$ . De acordo com o fluxo dos dados na rede, os vetores são utilizados em operações de multiplicação (*pointwise multiplication*, adição (*pointwise addition*) e concatenação (*vector concatenation*), junto com os pesos calculados (PHI, 2020). A lista completa de hiperparâmetros e seus valores padrões são encontradas em TensorFlow (2021d).

Figura 8 – Componentes de uma célula de LSTM



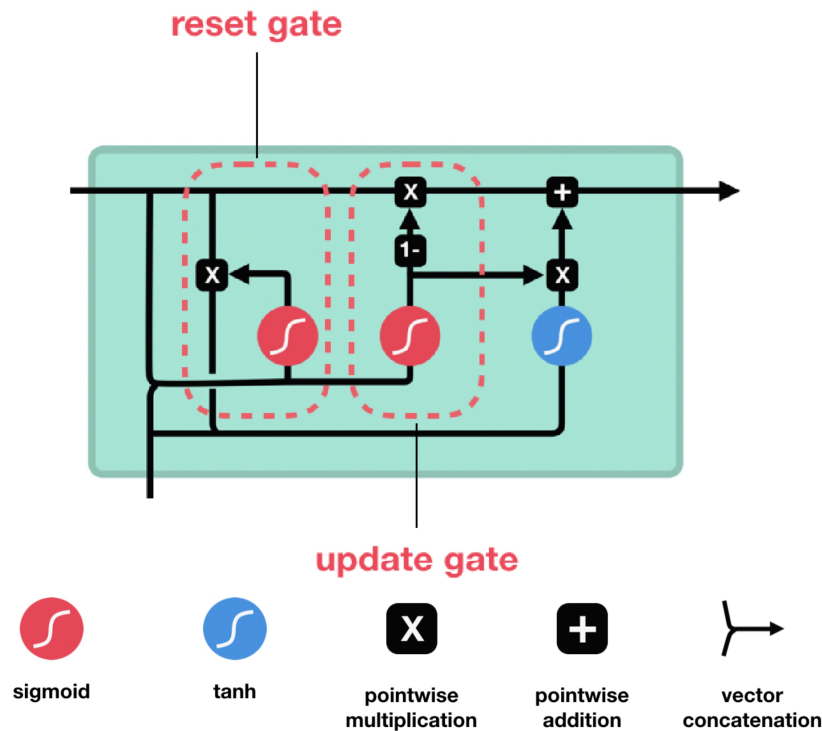
Fonte: Adaptado de (PHI, 2020)

### 3.5.2 Gated Recurrent Unit

A GRU consiste de um tipo de RNN com o mesmo propósito da LSTM, embora considerada uma melhoria em relação a anterior, utilizando menos portas e parâmetros (KUMAR; WIIL, 2019; BIANCHI et al., 2017). Em razão disso, redes GRU superaram a performance de redes LSTM em várias aplicações, treinando e convergindo mais rápido, se mostrando computacionalmente menos custosas e igualmente eficientes (KUMAR; WIIL, 2019; BIANCHI et al., 2017). Empregada também em classificação e previsão de séries temporais, este tipo de rede ajuda na recuperação de informações em escalas de tempo distintas (BIANCHI et al., 2017; KUMAR; WIIL, 2019).

Cada célula de uma GRU possui um estado de célula (memória) que utiliza de um estado escondido (*hidden state*) para transferir a informação, com é feito na LSTM. Contudo, a GRU utiliza de apenas duas portas, conforme ilustra a Figura 9: a porta de reinicialização (*reset gate*) e a porta de atualização (*update gate*). A porta de atualização opera de forma similar às portas de esquecimento e a de entrada de uma LSTM, decidindo quais informações descartar e quais informações novas adicionar a memória. A porta de reinicialização, por sua vez, é utilizada para decidir quais informações do passado devem ser esquecidas (PHI, 2020). A lista completa de hiperparâmetros e seus valores padrões são encontradas em TensorFlow (2021c).

Figura 9 – Componentes de uma célula de GRU

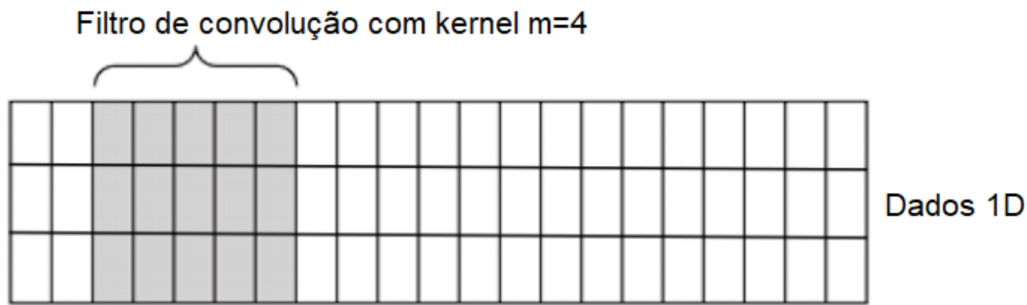


Fonte: Adaptado de (PHI, 2020)

### 3.5.3 Convolutional Neural Network

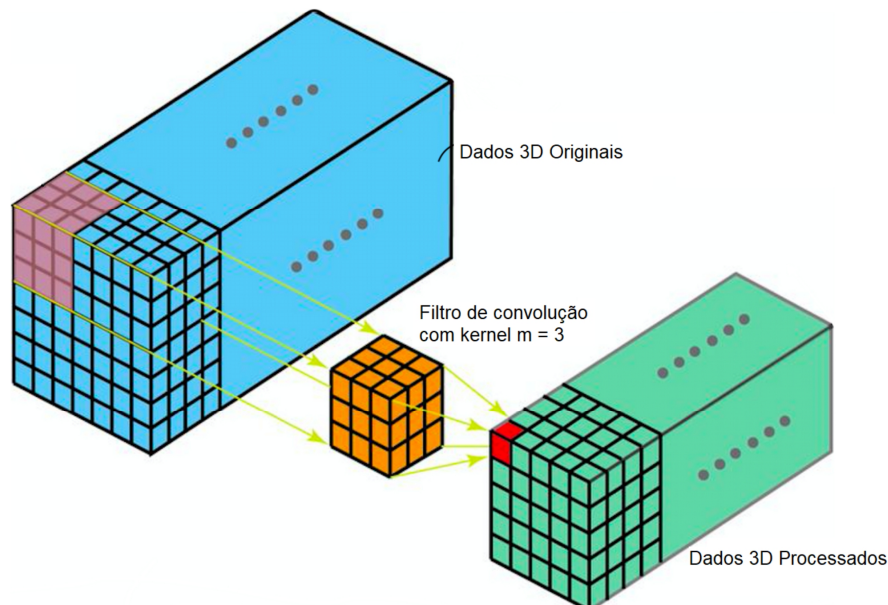
A CNN é um tipo de rede neural projetada para tratar eficientemente dados de imagem, se mostrando efetiva em aplicações de visão computacional, como classificação de imagem, localização de objetos, etc. (BROWNLEE, 2018). Através da aplicação de convoluções, este tipo de rede é capaz de aprender como extrair características de alto nível diretamente nos dados brutos, que representam bem os padrões a serem reconhecidos (DIXON et al., 2019; GOODFELLOW; BENGIO; COURVILLE, 2016). Esta abordagem, denominada *representation learning* (BROWNLEE, 2018), também pode ser aplicada em séries temporais, onde apresenta duas vantagens sobre outras técnicas: dependência local, uma vez que os sinais próximos provavelmente estão correlacionados; e invariância de escala para diferentes passos ou frequências (WANG et al., 2019). Neste tipo de rede, cada camada de convolução possui uma quantidade  $n$  de filtros, cada um deles com um *kernel* de tamanho  $m$ , conforme ilustra as Figuras 10 e 11. Após a convolução, geralmente existem camadas de *pooling* e *fully connected*, que executam tarefas de classificação (WANG et al., 2019). A lista completa de hiperparâmetros e seus valores padrões são encontradas em TensorFlow (2021a).

Figura 10 – Convolução em dados 1D



Fonte: Adaptado de (VERMA, 2020)

Figura 11 – Convolução em dados 3D



Fonte: Adaptado de (VERMA, 2020)

### 3.5.4 Convolutional Long Short-Term Memory

A ConvLSTM é um tipo de RNN variante da LSTM (SALMAN et al., 2018). Nestas redes existem operações de convolução dentro da célula de LSTM, de forma que as estruturas de convolução são aplicadas tanto na transição de entrada para estado, quanto nas transições de estado para estado. Desta forma, todas as operações de multiplicação de matrizes interna são substituídas por convoluções (RAHMAN; ADJEROH, 2019). Neste ponto se mostra evidente a diferença para a híbrida CNN-LSTM, nas quais estruturas de convolução (CNN) são aplicadas como a primeira camada e sequencialmente na segunda camada é aplicada LSTM (RAHMAN; ADJEROH, 2019). A lista completa de hiperparâmetros e seus valores padrões são encontradas em TensorFlow (2021b).

### 3.5.5 Convolutional Neural Network Long Short-Term Memory

A CNN-LSTM constitui um tipo de rede neural híbrida que utiliza ambas as camadas de convolução (CNN) e recorrência (LSTM) (DEEP; ZHENG, 2019). Nestas redes, os sinais são inicialmente agrupados em sequências temporais, e cada sequência é subdividida em blocos menores, as subsequências. Nestas subsequências são aplicadas camadas de convolução para extração de características complexas (DEEP; ZHENG, 2019). Em seguida, estas características são agrupadas novamente na sequência original e empregadas em camadas recorrentes, para interpretar a sequência temporal das características produzidas.

### 3.6 MÉTRICAS DE AVALIAÇÃO

Para mensurar os resultados dos experimentos foram adotadas métricas comumente utilizadas em problemas de classificação: acurácia, precisão, *recall* e *f1-score* (RODRIGUES, 2020; RODRIGUES, 2021; SHUNG, 2020). Em um problema de classificação, os resultados podem ser classificados em Verdadeiros Positivos (VP), sendo a classificação correta da classe Positivo; Verdadeiros Negativos (VN), a classificação correta da classe Negativo; Falsos Positivos (FP), erro em que o modelo previu a classe Positivo quando o valor real era classe Negativo; e Falsos Negativos (FN), erro em que o modelo previu a classe Negativo quando o valor real era classe Positivo (RODRIGUES, 2020). A Figura 12 ilustra os resultados possíveis.

Figura 12 – Matriz confusão de classificação

		Detectada	
		Sim	Não
Real	Sim	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Fonte: (RODRIGUES, 2020)

A acurácia indica uma performance geral do modelo. Desta forma, consiste da proporção de amostras classificadas corretamente em relação ao total de amostras, considerando todas as classes de dados. A fórmula desta métrica é apresentada a seguir:

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \quad (3.1)$$

Por outro lado, as métricas de precisão, *recall* e *f1-score* avaliam cada uma das classes de dados separadamente, de modo que se possa verificar problemas de desbalanceamento e viés. A precisão mede a proporção de amostras preditas corretamente em relação ao total de predições para determinada classe de dados, conforme a seguinte fórmula:

$$Precisão = \frac{VP}{VP + FP} \quad (3.2)$$

A *recall* mede a proporção de amostras preditas corretamente para determinada classe de dados em relação a todas as amostras reais daquela classe, conforme a seguinte fórmula:

$$Recall = \frac{VP}{VP + FN} \quad (3.3)$$

Por fim, a *f1-score* consiste da média harmônica de precisão e *recall*, conforme a fórmula a seguir:

$$F1-Score = \frac{2 \cdot precisão \cdot recall}{precisão + recall} \quad (3.4)$$

## 4 REVISÃO DA LITERATURA

Com o intuito de estabelecer o estado da arte acerca dos métodos de percepção veicular baseados em sinais de sensores inerciais, foi conduzida uma Revisão Sistemática da Literatura (RSL) guiada pelos procedimentos descritos em Kitchenham et al. (2009) e Biolchini et al. (2005). Esta revisão foi publicada em sua forma completa em Menegazzo e Wangenheim (2018), com a compilação de dados extraídos dos estudos selecionados e a descrição detalhada de cada trabalho em ordem cronológica de publicação. Baseado na revisão, foi publicado também em Menegazzo e Wangenheim (2020) um Mapeamento Sistemático da Literatura (MSL) das abordagens e métodos no estado da arte. Nesta seção, é apresentada a sistemática aplicada na busca e seleção de estudos primários, assim como detalhada a análise produzida e as lacunas de pesquisa encontradas, as quais se pretende explorar nesta pesquisa.

### 4.1 DEFINIÇÕES DA BUSCA

Na construção do protocolo que guiou a revisão sistemática, foram formuladas as seguintes **perguntas de pesquisa**:

1. Quais os métodos aplicados no processamento de sinais de sensores inerciais para gerar percepção veicular?
2. Quais são os padrões reconhecidos e suas classificações?
3. Quais as variáveis utilizadas e os fatores de dependência do modelo?
4. Quais sensores foram utilizados, suas características e configurações? Incluindo sensores não inerciais relacionados aos fatores de dependência.
5. Como os sensores foram distribuídos e posicionados no veículo?
6. Quais foram as aplicações finais das percepções veiculares desenvolvidas?

Com estas questões propostas, buscou-se identificar o contexto no qual a amostragem de sinais ocorreu, os sensores utilizados, suas configurações e colocação no veículo. Estas informações são muito importantes para verificar a sensibilidade dos sensores, taxas de amostragem, referenciais e condições gerais de análise. Nos dados amostrados, foram analisadas as etapas de processamento realizadas e quais métodos foram aplicados para filtragem de sinais, reorientação de eixos e reconhecimento de padrões. Adicionalmente aos métodos, foram identificados os padrões de percepção veicular, as variáveis dos modelos e seus fatores de dependência, os quais afetam diretamente a adaptabilidade e portabilidade da solução. Desta forma, o seguinte escopo de revisão foi definido:

**Objetivo:** Identificar e analisar métodos baseados em sensoriamento inercial para geração de percepção veicular.

**Controle:** Coleção de artigos relacionados ao objeto de pesquisa, obtidos por meio de revisão exploratória anterior.

**Intervenção:** Métodos e técnicas para reconhecimento e classificação de padrões em sinais unidimensionais.

**População:** Publicações relacionadas a projetos de sensoriamento veicular que usam acelerômetro ou giroscópio para reconhecer o ambiente em que o veículo está trafegando ou seu comportamento de condução.

**Resultados:** Lista de métodos e técnicas utilizadas no reconhecimento de padrões a partir de sinais inerciais para produzir percepção veicular.

**Aplicações:** Projetos de sensoriamento veicular em ITS, como agentes de navegação autônoma, ADAS, aplicativos de caixa preta, sistemas de *mobile crowdsensing* e outros.

Após definir o objeto de pesquisa e o escopo da revisão, foram selecionadas as fontes nas quais os estudos primários foram buscados. A seleção das fontes seguiu os critérios de estar disponível na Internet, ser banco de dados científicos da área e possuir mecanismos avançados de busca baseados na composição lógica de palavras-chave e restrições de campos de documentos. Neste sentido, três fontes foram selecionadas, detalhadas na Tabela 2.

Tabela 2 – Definições da busca

<b>Fontes</b>
<ol style="list-style-type: none"> <li>1. IEEE Xplore Digital Library</li> <li>2. ACM Digital Library</li> <li>3. Science Direct</li> </ol>
<b>Critérios de Inclusão</b>
<ol style="list-style-type: none"> <li>1. Artigos publicados e disponíveis integralmente em bases de dados científicas digitais.</li> <li>2. Trabalhos recentes publicados entre os anos de 2013 e 2019.</li> <li>3. Artigos publicados no idioma inglês.</li> <li>4. Artigos que reconhecem algum padrão de percepção veicular em sinais de sensores inerciais.</li> </ol>
<b>Critérios de Exclusão</b>
<ol style="list-style-type: none"> <li>1. Artigos que não estão disponíveis integralmente nas fontes pesquisadas.</li> <li>2. Artigos que não foram publicados entre os anos de 2013 e 2019.</li> <li>3. Artigos que não usam sensores inerciais para reconhecimento de padrões.</li> <li>4. Artigos que não usam sensores inerciais com três eixos.</li> <li>5. Artigos que não realizam reconhecimento da percepção veicular.</li> <li>6. Artigos de sensoriamento inercial para veículos de qualquer tração que não sejam mecânicos.</li> <li>7. Artigos que usam tecnologias intrusivas ou ativas.</li> <li>8. Artigos em outros idiomas, não incluídos no protocolo.</li> </ol>

Fonte: Desenvolvido pelo autor.

Para obter estudos primários a partir das bases de dados científicas definidas, com base nas palavras-chave obtidas nos estudos de controle foram construídas inicialmente as *strings* de



busca, detalhados nas Tabelas 3 e 4. Essas *strings* foram modeladas de acordo com as características de cada fonte. Posteriormente, as *strings* foram submetidas ao respectivo mecanismo de busca e seus resultados analisados por meio de uma estratégia de seleção. No processo de seleção, foram aceitos para a revisão os trabalhos que atenderam a todos os critérios de inclusão e a nenhum dos critérios de exclusão, detalhados na Tabela 2.

Tabela 3 – *Strings* de busca para exterocepção veicular

<b>Fonte</b>	IEEE Xplore Digital Library
<b>String de Busca</b>	((Road <b>OR</b> Pavement <b>OR</b> Pothole <b>OR</b> Hole <b>OR</b> Bump <b>OR</b> "Speed Breaker") <b>AND</b> (Accelerometer <b>OR</b> Gyroscope <b>OR</b> "Inertial Sensor"))
<b>Filtros</b>	Intervalo: 2013-2019. Busca apenas em metadados.
<b>Fonte</b>	ACM Digital Library
<b>String de Busca</b>	((Road <b>OR</b> Pavement <b>OR</b> Pothole <b>OR</b> Bump <b>OR</b> "Speed Breaker") <b>AND</b> (Accelerometer <b>OR</b> Gyroscope <b>OR</b> "Inertial Sensor"))
<b>Filtros</b>	Intervalo: 2013-2019. Busca em títulos e resumos.
<b>Fonte</b>	Science Direct
<b>String de Busca</b>	((Road <b>OR</b> Pavement <b>OR</b> Pothole <b>OR</b> Hole <b>OR</b> Bump <b>OR</b> "Speed Breaker") <b>AND</b> (Accelerometer <b>OR</b> Gyroscope <b>OR</b> "Inertial Sensor"))
<b>Filtros</b>	Intervalo: 2013-2019. Busca em títulos, resumos e palavras-chave.

Fonte: Desenvolvido pelo autor.

Tabela 4 – *Strings* de busca para propriocepção veicular

<b>Fonte</b>	IEEE Xplore Digital Library
<b>String de Busca</b>	("Driving Events" <b>OR</b> "Driving Patterns" <b>OR</b> "Driving Style" <b>OR</b> "Drive Behaviour") <b>AND</b> (Accelerometer <b>OR</b> Gyroscope <b>OR</b> "Inertial Sensor")
<b>Filtros</b>	Intervalo: 2013-2019. Busca apenas em metadados.
<b>Fonte</b>	ACM Digital Library
<b>String de Busca</b>	("Driving Events" <b>OR</b> "Driving Patterns" <b>OR</b> "Driving Style" <b>OR</b> "Drive Behaviour") <b>AND</b> (Accelerometer <b>OR</b> Gyroscope <b>OR</b> "Inertial Sensor")
<b>Filtros</b>	Intervalo: 2013-2019. Busca em títulos e resumos.
<b>Fonte</b>	Science Direct
<b>String de Busca</b>	("Driving Events" <b>OR</b> "Driving Patterns" <b>OR</b> "Driving Style" <b>OR</b> "Drive Behaviour") <b>AND</b> (Accelerometer <b>OR</b> Gyroscope <b>OR</b> "Inertial Sensor")
<b>Filtros</b>	Intervalo: 2013-2019. Busca em títulos, resumos e palavras-chave.

Fonte: Desenvolvido pelo autor.

## 4.2 EXECUÇÃO DA BUSCA

Com as definições estabelecidas, foram realizadas buscas por estudos primários nas três fontes selecionadas. Devido à limitação de número de termos nos mecanismos de busca, foram realizadas duas buscas, uma para exterocepção e outra para propriocepção, conforme as *strings* de busca construídas. A busca por exterocepção resultou em 737 artigos e a propriocepção em 239 artigos. Após análise dos estudos de acordo com os critérios estabelecidos, 66 artigos sobre exterocepção e 25 sobre propriocepção foram selecionados para esta revisão, conforme detalhado na Tabela 5.

Tabela 5 – Resultados sumarizados das buscas

Fonte	Exterocepção		Propriocepção	
	Recuperado	Selecionado	Recuperado	Selecionado
IEEE Xplore Digital Library	619	51	196	21
ACM Digital Library	32	6	6	1
Science Direct	86	9	37	3
<b>Total</b>	<b>737</b>	<b>66</b>	<b>239</b>	<b>25</b>

Fonte: Desenvolvido pelo autor.

## 4.3 ANÁLISE DO ESTADO DA ARTE

Nesta seção são apresentados uma análise multiaspecto e um mapeamento dos estudos analisados, principalmente em relação à abordagem, métodos e técnicas empregadas. Consideramos para esta revisão as etapas de coleta de dados, pré-processamento e processamento. Entretanto, existe também a etapa de pós-processamento, geralmente empregada com foco na aplicação final dos dados e métodos para melhorar a confiabilidade das percepções. Sendo assim, esta etapa não se mostra essencial nesta revisão, já que buscamos a confiabilidade nas três primeiras etapas, através de uma modelagem dos fatores de dependência.

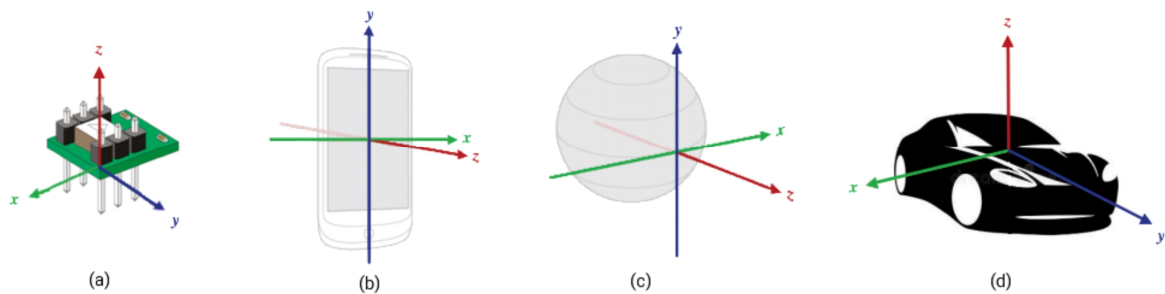
Nas próximas subseções são detalhadas cada uma das etapas analisadas. Inicialmente são explicados os referenciais para amostragem e análise de dados, juntamente com as técnicas de reorientação de eixos, que envolvem as etapas de coleta e pré-processamento dos dados. A seguir, são apresentados os fatores de dependência que interferem nos sinais amostrados, relacionados às propriedades sensoriais, veiculares, de condução e ambientais. Na sequência, detalhamos as três etapas de análise. Na etapa de coleta de dados, são especificadas as plataformas de *hardware* utilizadas, a colocação dos sensores e as configurações para amostragem do sinal. Na etapa de pré-processamento, são apresentados os métodos aplicados para remoção de ruído, suavização de sinal, extração de características e segmentação de dados. Na etapa de processamento, são discutidas as técnicas utilizadas para reconhecer as exterocepções e as

propriocepções. Em seguida, são detalhados os padrões de percepção e suas classificações possíveis de serem gerados através de sensores inerciais, assim como suas aplicações finais. Na sequência, é apresentada a discussão e considerações acerca do estado da arte. Por fim, é discorrida uma análise comparativa detalhada do estado da arte das percepções específicas trabalhadas nesta pesquisa.

### 4.3.1 Referenciais

Os sensores inerciais não requerem nenhum referencial externo. Sendo assim, possuem seu próprio referencial, onde o sistema de coordenadas no qual os dados são amostrados é definido em relação a si próprios. No entanto, a análise dos dados com o objetivo de produzir percepção veicular deve ser realizada no referencial do veículo, como mostra a Figura 13. Para isso, é necessário reorientar os eixos, mapeando os dados brutos de um sistema para outro.

Figura 13 – Referenciais: (a) Referencial do sensor. (b) Referencial do sensor interno a dispositivos móveis. (c) Referencial da Terra. (d) Referencial do veículo.



Fonte: Desenvolvido pelo autor.

A orientação dos dados é importante no processo de análise para que se possa utilizar os valores de acordo com o tipo de percepção a ser realizada. Sendo assim, baseando-se no referencial do veículo, os padrões de percepção de ambiente são geralmente mais evidentes na aceleração do eixo Z e na taxa de rotação dos eixos X e Y. Já quanto aos padrões de propriocepção, mostram-se mais evidentes nos dados de aceleração nos eixos X e Y e na taxa de rotação do eixo Z. Para uma padronização de nomenclatura neste trabalho, serão usados os termos definidos nas Tabelas 6 e 7 para se referir aos dados em relação a cada um dos referenciais.

Tabela 6 – Referenciais para os dados de aceleração

Axis	Sensor	Veículo	Mundo
X	Aceleração em X	Aceleração Lateral	Aceleração Leste
Y	Aceleração em Y	Aceleração Longitudinal	Aceleração Norte
Z	Aceleração em Z	Aceleração Vertical	Aceleração Vertical (Mundo)

Fonte: Desenvolvido pelo autor.

Tabela 7 – Referenciais para os dados de taxa de rotação

Axis	Sensor	Veículo	Mundo
X	Taxa de Rotação Pitch	Taxa de Rotação Lateral	N/A
Y	Taxa de Rotação Roll	Taxa de Rotação Longitudinal	N/A
Z	Taxa de Rotação Yaw	Taxa de Rotação Vertical	N/A

Fonte: Desenvolvido pelo autor.

Para a reorientação dos eixos, duas estratégias foram identificadas nos estudos revisados: o posicionamento controlado e o mapeamento através de fórmulas baseadas nos ângulos de Euler. O posicionamento controlado consiste em uma técnica simples, onde sensor é colocado no veículo de forma que os eixos nos referenciais coincidam, ou seja, os eixos do sensor ficam alinhados com os eixos do veículo. Portanto, não é necessário aplicar pré-processamento para reorientação. Essa técnica é usada tanto para os dados do acelerômetro e do giroscópio.

Os ângulos de Euler, por sua vez, aplicados apenas aos dados de aceleração, fornecem um meio de representar a orientação espacial tridimensional de qualquer referencial como uma composição de três rotações elementares. A orientação de referência pode ser tomada como uma orientação inicial a partir da qual o sistema de coordenadas gira para alcançar sua orientação real (SINGH et al., 2017). Assim, essas fórmulas reorientam os dados brutos usando os ângulos fornecidos. Estes dois métodos são detalhados, analisados e comparados nas subseções de coleta de dados e pré-processamento.

#### 4.3.2 Fatores de Dependência

Os valores amostrados através dos sensores inerciais, embora não dependam de um referencial externo para sua produção, são afetados por propriedades externas e internas aos sensores. Essas propriedades constituem fatores de dependência, que influenciam diretamente a corretude, amplitude e dispersão dos valores medidos. Através da análise da literatura e de experimentos exploratórios conduzidos, foi possível observar a existência de fatores de dependência, classificados na forma de quatro propriedades relacionadas às variações contextuais discutidas abaixo. Para que uma solução desenvolvida possa ser adaptativa na forma que está sendo proposta, é necessário que o modelo considere todas estas propriedades no seu desenvolvimento.

**Propriedades Sensoriais:** Quatro fatores de dependência estão relacionados às configurações e características dos sensores, sendo eles a orientação (KUMAR et al., 2017; ALAM et al., 2020), resolução, faixa de medição e taxa de amostragem. A orientação do sensor, conforme discutido na seção anterior, afeta a amostragem de dados no sistema de coordenadas correto. Portanto, o sensor deve ser colocado de forma a ficar alinhado com o sistema de coordenadas do veículo ou ter uma reorientação de eixos para esse referencial.

Isso implica na confiabilidade dos dados existentes em cada eixo de análise, pois cada um deles visa atingir certa percepção veicular.

O segundo e o terceiro fator, resolução e faixa de medição, estão fortemente correlacionados. A resolução é definida de acordo com a faixa de medição FSR escolhida para o sensor. Portanto, é necessário que o sensor tenha uma faixa de medição adequada para poder amostrar os dados sem saturar, ou seja, que a faixa de medição do sensor não seja menor que os valores possíveis a serem monitorados. A resolução, estabelecida a partir da faixa de valores a serem analisados, fornece o quão próximo o valor medido é comparado ao valor real, no processo de quantização e representação no sistema binário. Assim, devido à limitação do número de bits representado pelos sensores, quanto maior a faixa de medição, menor a resolução.

Por fim, a taxa de amostragem descreve a frequência da coleta de dados por segundo. A escolha desse valor deve levar em consideração não apenas o custo computacional de armazenamento e processamento dos dados amostrados, mas também se, a uma determinada velocidade, será possível obter amostras suficientes para realizar as percepções. Dessa forma, quando o veículo está em alta velocidade, percepções transientes, como buracos, precisam de uma taxa de amostragem satisfatória, para que seja possível adquirir amostras suficientes desse evento.

**Propriedades Veiculares:** Relacionam-se a estrutura veicular. O principal fator de dependência é o sistema de suspensão (KUMAR et al., 2017; WICKRAMARATHNE; GARG; BAUER, 2018; ALAM et al., 2020). Este sistema, com o objetivo de amortecer os impactos causados pelas irregularidades e obstáculos na superfície da pista, faz com que os valores medidos sejam reduzidos dependendo da localização dos sensores na infraestrutura veicular. Desta forma, os valores medidos abaixo da suspensão apresentam interferência da massa não suspensa através da rigidez e absorção do pneu, e os medidos acima da suspensão adicionam também a influência da massa suspensa, através da mola e amortecedor (AL-YAFEAI; DARABSEH; MOURAD, 2019). Desta forma, para poder realizar as percepções em diferentes veículos é necessário considerar que cada um deles possui uma estrutura diferente, com diferentes sistemas de suspensão, e que diferentes colocações do sensor em um mesmo veículo podem ter mais ou menos interferência destas propriedades.

**Propriedades de Condução:** As propriedades de condução, intrinsecamente vinculadas as propriedades veiculares, relacionam-se aos diferentes estilos de condução do veículo. Cada motorista possui seu estilo de condução, aplicando mais ou menos aceleração, dirigindo velocidades mais altas ou mais baixas, realizando diferentes manobras (ALAM et al., 2020). Cada um destes fatores impacta nos sinais amostrados, de acordo com a tração e direção aplicada e o modo com que o veículo interage com o ambiente. O principal fator de dependência é a velocidade longitudinal (BRUNAUER; REHRL, 2016; DOU-

ANGPHACHANH; ONEYAMA, 2013; GUETA; SATO, 2017; KUMAR et al., 2017; LIMA et al., 2016; M.; GOPI, 2017; NALAVDE; SURVE, 2015; SINGH et al., 2017; ALAM et al., 2020), a qual possui duas principais implicações. Uma vez que as curvas são feitas em todos os eixos do sistema de coordenadas do veículo, ocorre a produção do componente de força centrífuga. Portanto, as acelerações medidas dependem diretamente da velocidade aplicada. A segunda implicação da velocidade está na distribuição no tempo dos valores amostrados. Em velocidades mais baixas, mais amostras do evento são coletadas e vice-versa. Assim, com o veículo em velocidades diferentes, diferentes quantidades de amostras são obtidas para um dado evento.

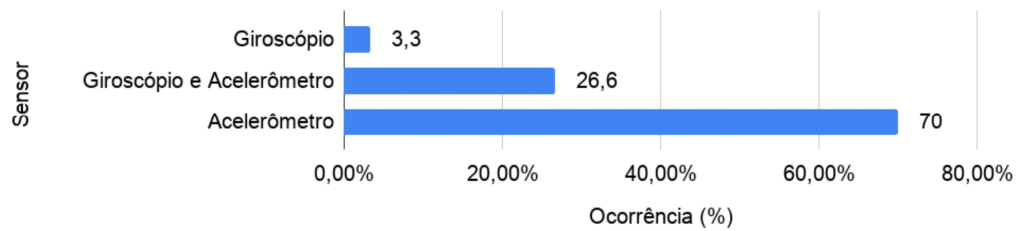
**Propriedades Ambientais:** Estas propriedades relacionam-se com as características do ambiente externo ao veículo (ALAM et al., 2020), as quais afetam diretamente os valores amostrados, de forma que uma mesma percepção possa ter padrões muito diferentes. Na classificação de tipo de superfície de pista, um determinado tipo de pavimento pode apresentar variações no estado de conservação, possuir buracos, lombadas, outras anomalias e obstáculos. A variação destas características não deve influenciar a classificação final. Na detecção de lombadas este tipo de obstáculo pode estar presente em pistas com diferentes pavimentações, como asfalto e paralelepípedo (MASINO et al., 2017). Cada um destes tipos de pavimento apresenta diferentes padrões nos sinais nos sensores. Além disso, as lombadas podem apresentar variações em suas dimensões, seja pela forma como foram construídas ou pela falta de manutenção. Independente destas condições, estas variações não devem influenciar a classificação final, onde lombadas diferentes devem ser reconhecidas da mesma maneira.

### 4.3.3 Coleta de Dados

A coleta de dados nos estudos analisados foi realizada em diferentes contextos, de acordo com o tipo de percepção que se pretendia produzir. Esses contextos envolviam estradas com diferentes pavimentos, estados de conservação, condições climáticas, irregularidades, obstáculos e modos de condução. De acordo com Figura 14, observamos que a maioria dos estudos utilizou apenas os dados do sensor acelerômetro, uma vez que este é mais facilmente encontrado em dispositivos móveis do que o giroscópio. No entanto, outras soluções foram propostas utilizando dados do giroscópio ou de ambos, embora em quantidade consideravelmente menor.

Os estudos também foram agrupados de acordo com a plataforma utilizada, detalhada na Figura 15. Aproximadamente 83% dos estudos utilizaram sensores inerciais embarcados em dispositivos móveis como *smartphones* e *tablets*, enquanto que apenas 17% utilizaram anexo aos veículos. No uso de sensores em dispositivos móveis, as soluções propostas são mais vulneráveis a erros, pois os dispositivos não permanecem fixos o tempo todo, dependem de uma reorientação dos eixos e, geralmente, possuem sensores menos robustos. Em geral, essa abordagem tem sido utilizada em sistemas de *mobile crowdsensing*, onde os dados produzidos

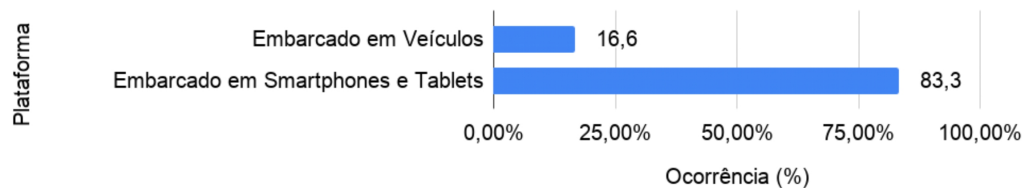
Figura 14 – Sensores inerciais utilizados nos estudos revisados



Fonte: Desenvolvido pelo autor.

localmente são enviados para um servidor remoto, que recebe percepções veiculares de várias fontes. Após o pós-processamento para melhorar a confiabilidade dos dados, as percepções retornam aos usuários em aplicativos de assistência ao motorista. Por sua vez, o uso de sensores anexados aos veículos foi combinado com a utilização de SBC Raspberry Pi, ou microcontrolador, como o Arduíno, visando principalmente resultados locais com maior precisão e acurácia, sem a necessidade de processamento remoto ou dados de outras fontes.

Figura 15 – Plataformas utilizadas nos estudos revisados



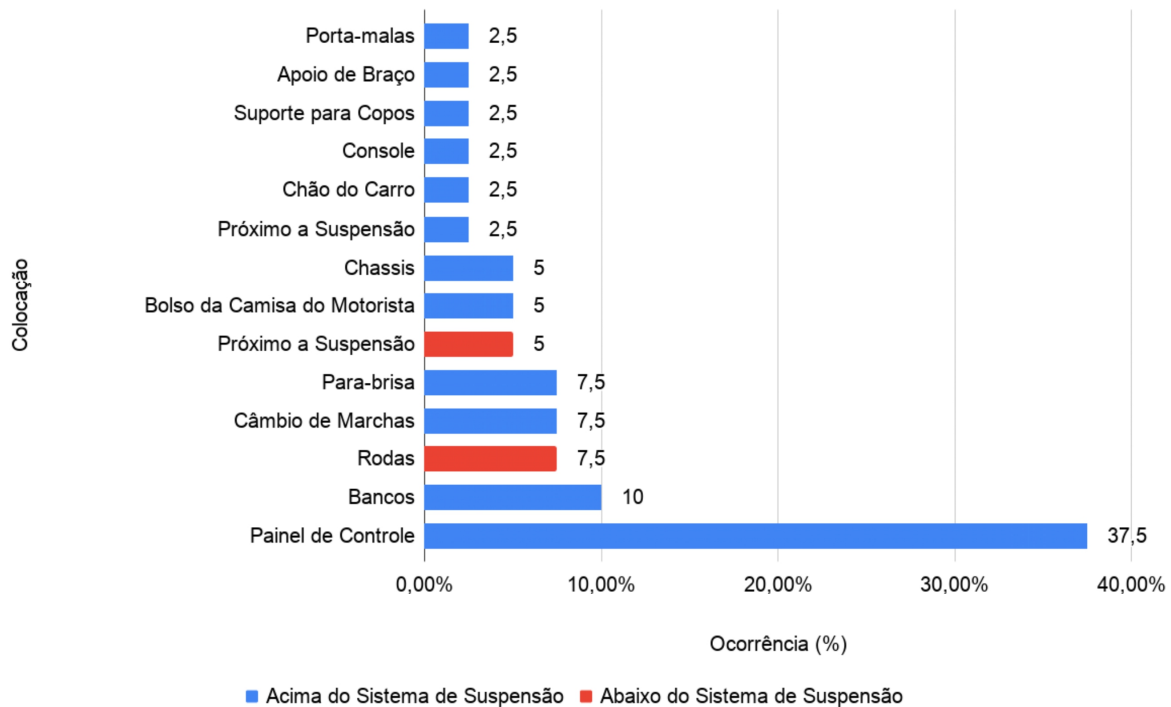
Fonte: Desenvolvido pelo autor.

Os sensores foram colocados no veículo de acordo com a plataforma utilizada. Sensores inerciais em dispositivos móveis foram colocados dentro da cabine, geralmente no painel de controle, bancos, próximo ao câmbio de marchas ou fixado com suporte no para-brisas. Sensores embarcados no veículo foram colocados no painel, no chassi, nas rodas, próximo e abaixo da suspensão e próximo e acima da suspensão. A Figura 16 detalha as colocações encontradas. Devido à economia de recursos, especialmente a bateria, os dados dos sensores de dispositivos móveis foram coletados a uma taxa de amostragem na faixa de  $Hz$ , enquanto os fixados no veículo utilizaram uma amostragem maior, na faixa de  $KHz$ . Além dos sensores inerciais, os estudos também utilizaram fontes de dados auxiliares, como magnetômetros, GPS e *On-Board Diagnostic II* (OBD-II).

#### 4.3.4 Pré-Processamento

Após a coleta, os dados brutos foram pré-processados para serem utilizados nos modelos que reconhecem e classificam os padrões de percepção. O pré-processamento foi subdividido em três fases, de acordo com sua finalidade. Na primeira, foram aplicados métodos para a reorientação de eixos. Na segunda, os dados foram ajustados para remoção de ruído, suaviza-

Figura 16 – Colocação dos sensores no veículo



Fonte: Desenvolvido pelo autor.

ção de sinal, segmentação de dados e extração de características. Finalmente, na terceira fase, foram aplicados métodos para georreferenciamento dos dados.

Na primeira etapa de pré-processamento, poucos estudos utilizaram a reorientação dos eixos entre os sistemas de coordenadas, mapeando os dados entre os referenciais de coleta e análise. A maioria utilizou os dados no referencial amostrado, com os sensores posicionados de forma arbitrária. Dentre os que utilizaram reorientação de eixos, a maioria empregou fórmulas baseadas nos ângulos de Euler (LI; GOLDBERG, 2018; ORHAN; EREN, 2013; SINGH et al., 2017; VITTORIO et al., 2014; VLAHOGIANNI; BARMPOUNAKIS, 2017). Nesta técnica, os ângulos foram calculados de diferentes maneiras. Em Singh et al. (2017), Orhan e Eren (2013), Vittorio et al. (2014) o referencial inicial foi estabelecido como estado estacionário ( $x = 0m/s^2$ ,  $y = 0m/s^2$  e  $z = 9,81m/s^2$ ), a partir do qual os ângulos de mapeamento foram calculados. Em Li e Goldberg (2018) os valores do acelerômetro foram definidos como estado inicial, medindo os ângulos continuamente. Em Singh et al. (2018) como estado inicial foram utilizadas as médias individuais dos valores de aceleração nos três eixos, dos últimos 15 segundos, quando a magnitude total estava próxima a  $9,8 \pm 0.2m/s^2$ . Todos eles empregaram como orientação final os valores atuais de aceleração.

Nos estudos analisados, os ângulos de Euler foram calculados com base na localização da gravidade, reorientando o eixo Z perpendicular ao plano do solo, apontando em direção ao céu. Para adicionar uma orientação aos dados dos eixos X e Y, alguns estudos utilizaram do

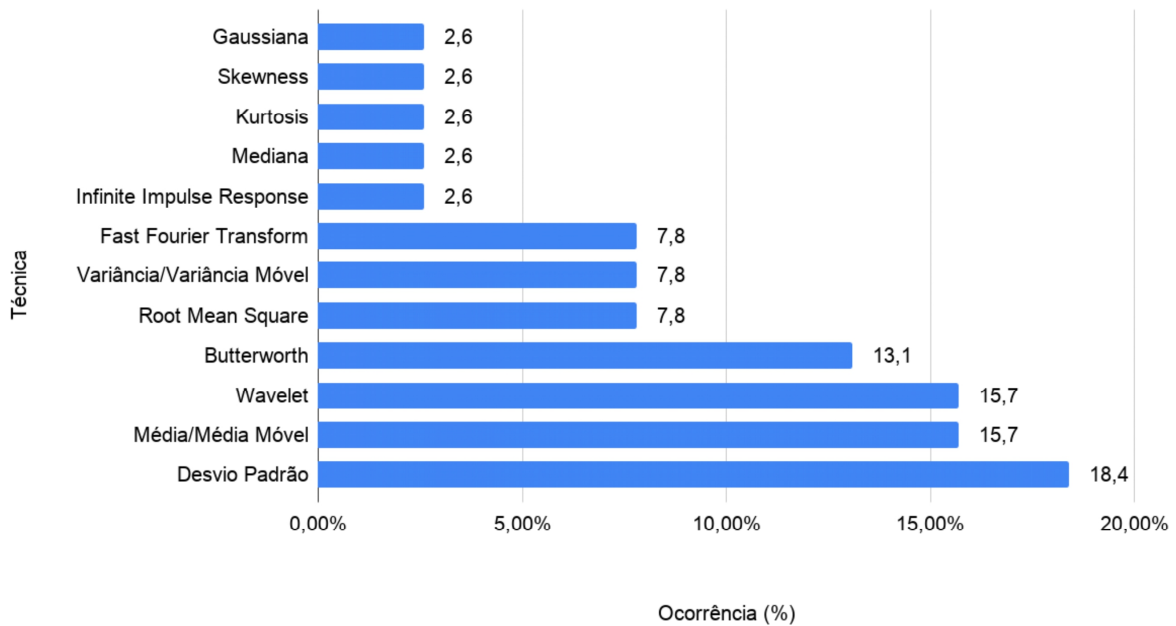


sensor magnetômetro, que mede o campo geomagnético ambiental (SATTAR; LI; CHAPMAN, 2018). Desta forma, o eixo X se torna tangencial ao solo e aponta para o leste, enquanto o eixo Y também tangencial ao solo aponta para o polo norte geomagnético (SATTAR; LI; CHAPMAN, 2018). Alguns estudos anteriores também utilizaram o sensor de gravidade para cálculo da orientação inicial e final. Este sensor baseado em *software* deriva seus dados através da fusão de dados do acelerômetro e do giroscópio, medindo a força gravitacional distribuída em cada um dos eixos físicos. Em suma, a utilização das fórmulas baseadas em ângulos de Euler mapeou os dados do acelerômetro para o referencial da terra, o qual sobrepõe apenas o eixo Z do referencial do veículo, e somente em situações de quando o veículo está em terreno plano.

Na segunda etapa de pré-processamento, várias técnicas foram aplicadas para ajustes nos sinais, mostradas na Figura 17, utilizadas principalmente no domínio do tempo, embora outros domínios como frequência e tempo-frequência também tenham sido explorados, conforme detalha a Figura 18. Abordagens estatísticas como Gaussiana (POOJA; HARIHARAN, 2017), *Skewness* (ALQUDAH; SABABHA, 2016), *Kurtosis* (ALQUDAH; SABABHA, 2016), Mediana (ALQUDAH; SABABHA, 2016; LI et al., 2016), *Root Mean Square* (RMS) (JANG et al., 2015; LI; GOLDBERG, 2018; SHARMA et al., 2015), Média/Média Móvel (ALQUDAH; SABABHA, 2016; ANDRIA et al., 2016; BOSE et al., 2018; LI et al., 2016; PHOLPRASIT; CHOCHAIWATTANA; SAIPRASERT, 2015; SAVERA et al., 2016; SINGH et al., 2017), Variância/Variância Móvel (ALQUDAH; SABABHA, 2016; ANDRIA et al., 2016) e Desvio Padrão (ANDRIA et al., 2016; BELLO-SALAU et al., 2018; HOU et al., 2017; LIMA et al., 2016; PHOLPRASIT; CHOCHAIWATTANA; SAIPRASERT, 2015; PRAPULLA; RAO; HERUR, 2017) foram utilizados no domínio do tempo para suavizar os sinais e extrair características dos dados de vibração. Entre as pesquisas que utilizaram pré-processamento de sinal, cerca de 60% deles utilizaram métodos estatísticos. No domínio da frequência, os filtros *Infinite Impulse Response* (IIR) (WU et al., 2018) e *Butterworth* (HOU et al., 2017; NISKANEN; TUONONEN, 2015; PITOŇÁK; FILIPOVSKY, 2016; SOUZA, 2018; WU et al., 2013) foram utilizados na remoção de componentes dos sinais. Já a *Fast Fourier Transform* (FFT) (ALLOUCH et al., 2017; DOUANGPHACHANH; ONEYAMA, 2013; DOUANGPHACHANH; ONEYAMA, 2014a) foi empregado para transformação de domínio, e *Wavelets* (BELLO-SALAU et al., 2018; EFTEKHARI; GHATEE, 2018; EL-WAKEEL et al., 2018; GUETA; SATO, 2017; SINGH et al., 2017; WANG et al., 2018) foram utilizadas na remoção de ruídos e extração de características. A segmentação dos dados foi realizada em janelas deslizantes em função do tempo (WANG et al., 2018), número de amostras (SINGH et al., 2017) ou distância (LI; GOLDBERG, 2018).

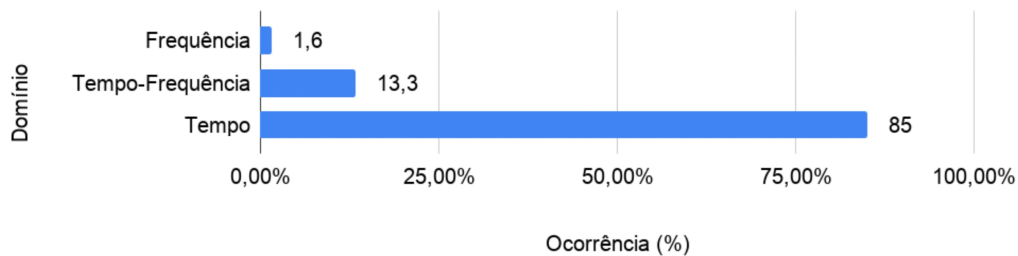
Na última fase de pré-processamento, uma vez que a taxa de amostragem de dados dos sensores inerciais (até  $1\text{KHz}$ ) é muito mais rápida que a amostragem de dados de localização ( $1\text{Hz}$ ), são necessárias técnicas para estimar a localização e velocidade antes da próxima amostra do GPS. Para isso, apenas Li e Goldberg (2018) utilizou um tratamento, aplicando interpolação linear com os dados de aceleração para obter dados de localização com uma frequência maior.

Figura 17 – Métodos utilizados no pré-processamento de sinais dos estudos revisados



Fonte: Desenvolvido pelo autor.

Figura 18 – Domínios de análise utilizados nos estudos revisados

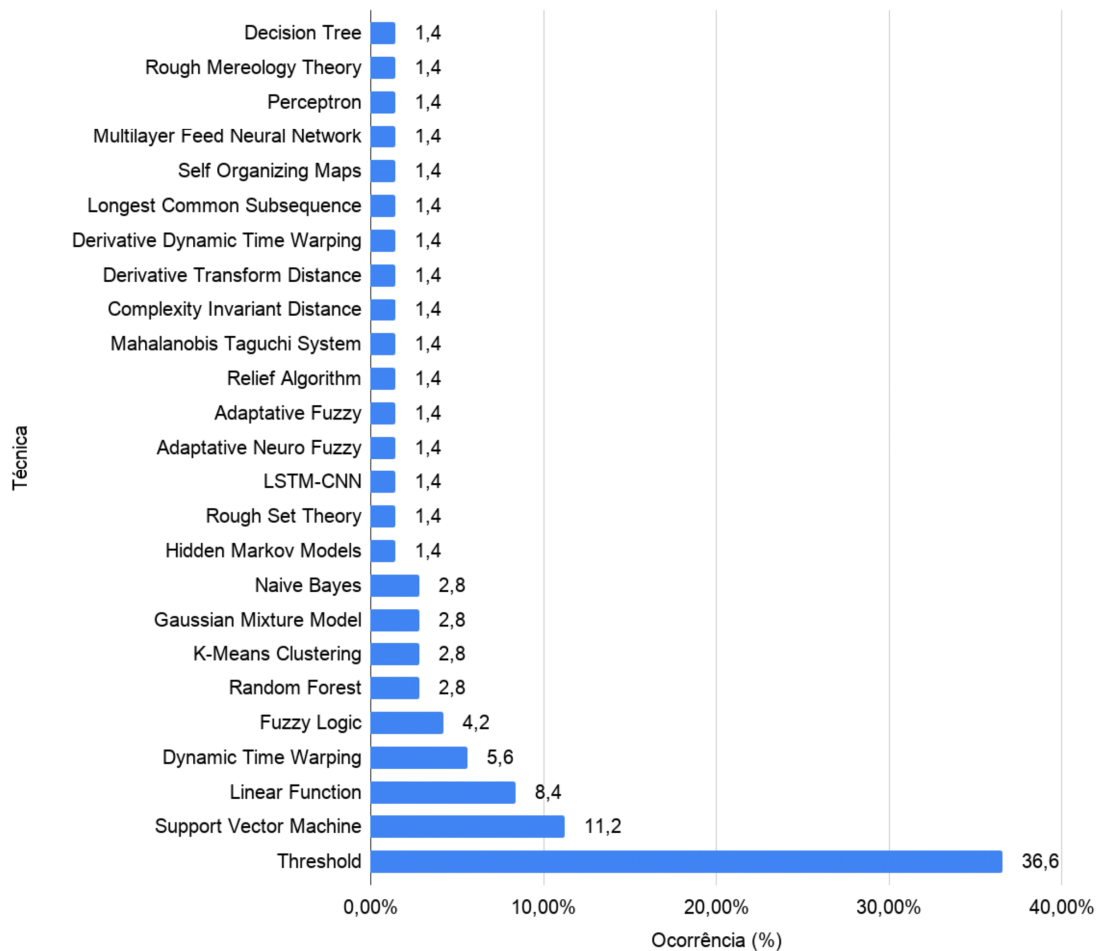


Fonte: Desenvolvido pelo autor.

#### 4.3.5 Processamento

Para reconhecer e classificar os padrões nos sinais pré-processados dos sensores inerciais, várias técnicas foram propostas, conforme detalhado na Figura 19. Essas técnicas foram classificadas de acordo com sua abordagem. A maioria dos estudos analisados utilizou técnicas simples baseadas em limiares ou funções lineares. Técnicas mais elaboradas também foram propostas, como métodos de alinhamento de séries temporais, aprendizado de máquina e modelos com conhecimento especializado.

Figura 19 – Técnicas utilizadas para reconhecimento e classificação de padrões de percepção veicular



Fonte: Desenvolvido pelo autor.

#### 4.3.5.1 Abordagens Baseadas em Limiares

Os métodos baseados em limiares (*threshold*) são extremamente simples e consistem na técnica mais utilizada nos estudos revisados. Esta abordagem é baseada na comparação dos valores pré-processados dos sensores com limiares estáticos ou dinâmicos. Alguns trabalhos utilizaram mais de um valor de comparação, construindo assim uma heurística baseada em limiares. Esses valores de comparação, em geral, são obtidos por meio de análises exploratórias dos sinais dos sensores inerciais, sem muito rigor metodológico.

Na aplicação de limiares para o reconhecimento de exterocepção, foram identificados eventos transientes e persistentes. No estudo de Orhan e Eren (2013) e Strutu, Stamatescu e Popescu (2013) foram identificados buracos e redutores de velocidade. Akhtar, Pandey e Gupta (2014), Astarita et al. (2014), Kaur et al. (2017) e Vittorio et al. (2014) reconheceram irregularidades do pavimento de forma genérica, sem classificar os padrões. Jamakhandi e Srinivasa (2014) detectou lombadas, poças e elevações abruptas. Bello-Salau et al. (2018), Gunawan,

Yanfi e Soewito (2015), Ghadge, Pandey e Kalbande (2015), Kumar et al. (2017), Li e Goldberg (2018), Pooja e Hariharan (2017) e Rishiwal e Khan (2016) identificaram solavancos/saliências e buracos, com Ghadge, Pandey e Kalbande (2015) validando o limiar com KMC e *Random Forest*. Gawad, Mougy e El-Meligy (2016) treinou um *Perceptron* para gerar o limiar, posteriormente empregado na identificação de anomalias nas estradas. Afrin, Mahmud e Razzaque (2015) identificou redutores de velocidade. Hsu, Perng e Wu (2016) reconheceu buracos e identificou a qualidade da estrada, classificando em irregular, áspera ou lisa. Lima et al. (2016) classificou a qualidade do pavimento em boa, normal, ruim e péssima, enquanto Li e Goldberg (2018) classificou em excelente/boa, regular, ruim, muito ruim/terrível.

Com relação à propriocepção, Niskanen e Tuononen (2015) usou uma abordagem de limiares para medir o contato do pneu com a pista e com água e, então, detectou aquaplanagem. Matilainen e Tuononen (2015) através o contato do pneu identificou a pista entre estradas secas e molhadas, informações úteis para o reconhecimento de aquaplanagem. Hou et al. (2017) detectou derrapagem. Sharma et al. (2015) identificou acelerações ou frenagens abruptas, curvas fechadas e mudanças de faixa frequentes. Li et al. (2016) detectou aceleração ou desaceleração anormal. Andria et al. (2016) identificou mudança de marcha: troca para marcha maior (suave ou forte) e redução de marcha (frenagem hidráulica ou frenagem com motor), além do estilo de direção: direção moderada ou agressiva. Finalmente, Găvrută, Marcu e Bogdan (2018) identificou curvas, acelerações e frenagens.

#### 4.3.5.2 *Abordagens de Alinhamento de Séries Temporais*

Outra abordagem amplamente utilizada nos artigos foram as técnicas de alinhamento de séries temporais. Nessa abordagem, os estudos visam encontrar o alinhamento ótimo entre duas sequências de valores dependentes do tempo (MÜLLER, 2007), para verificar a similaridade. Assim, dada uma série temporal que define um padrão de percepção, essa série é alinhada com os valores atuais do sensor para tentar reconhecer o padrão novamente. Na exterocepção, Alqudah e Sababha (2016) utilizou a técnica *Dynamic Time Warping* (DTW) para detectar anomalias na pista. Singh et al. (2017) usou a mesma técnica para reconhecer buracos e solavancos/saliências. Souza (2018) utilizou as técnicas DTW, *Longest Common Subsequence Similarity* (LCSS), *Derivative Dynamic Time Warping* (DDDTW), *Derivative Transform Distance* (DTD) e *Complexity Invariant Distance* (CID) para estabelecer a regularidade do pavimento asfáltico: regular ou deteriorado; tipo de superfície: asfalto, paralelepípedo e estradas de terra; e obstáculos no asfalto: lombada, remendo vertical, marcadores de elevação no pavimento e faixa de pedestres elevada. Na propriocepção, Pholprasit, Choochaiwattana e Saiprasert (2015) utilizou o DTW para reconhecer frenagem, aceleração, virando à esquerda ou à direita, mudança de faixa à esquerda e à direita, além do perfil de direção classificado em não agressivo e agressivo.

#### 4.3.5.3 *Abordagens de Funções Lineares*

As abordagens de função linear que identificamos visam mapear os padrões de vibração em índices de qualidade de estradas ou mapear eventos de condução em perfis de comportamento de direção. Prapulla, Rao e Herur (2017) e Bello-Salau et al. (2018) estabeleceram um índice personalizado de qualidade sobre a condição da pista. (BRUNAUER; REHRL, 2016) criou um índice de qualidade *Road Surface Condition Index* (RSCI). Chen et al. (2013), Douangphachanh e Oneyama (2013), Douangphachanh e Oneyama (2014a), Li e Goldberg (2018), Pitoňák e Filipovsky (2016), Singh et al. (2017) e Tomiyama e Kawamura (2016) mediram o Índice Internacional de Rugosidade (*International Roughness Index* - IRI). Em Chen et al. (2013) foi calculado um índice de qualidade chamado *Riding Quality Index* (RQI). Por meio do RQI, os valores foram mapeados para os conceitos de irregularidade, sendo eles: excelente, bom/competente e não competente. Saiprasert et al. (2014) mapeou eventos de direção para identificar o perfil de comportamento do motorista como muito seguro, seguro, agressivo ou muito agressivo.

#### 4.3.5.4 *Outras Abordagens*

Os demais trabalhos analisados propuseram um conjunto amplamente variado de abordagens, desde técnicas de caixa branca com inserção de conhecimento especializado até abordagens de caixa preta que empregam aprendizado de máquina. Na produção de exterocepção, Jang et al. (2015) usou uma *Multilayer Feed Forward Neural Network* para classificar a estrada em classe de abrupta, irregular ou regular/suave. Em Selmanaj, Corno e Savaresi (2014) foi utilizado de *Self Organizing Maps* (SOM) para classificar a estrada em *baseline*, irregular ou perigosa. Khaleghian e Taheri (2017) aplicou *Fuzzy Logic* para reconhecer o tipo de superfície em grama, terra, concreto ou asfalto. Fouad et al. (2014) usou *Rough Mereology Theory* para identificar redutores de velocidade. Allouch et al. (2017) utilizou *Decision Trees*, SVM e *Naive Bayes*, e Singh et al. (2018) usou uma SVM, ambos para identificar as condições da estrada de forma genérica. Nalavde e Surve (2015) aplicou KMC e SVM para classificar a estrada em acidentada ou suave. Savera et al. (2016) usou SVM para detectar lombadas e valas, enquanto Gueta e Sato (2017) com a mesma técnica identificou afundamentos, remendos, solavancos/saliências e buracos. El-Wakeel et al. (2018) usou SVM para reconhecer estradas suaves, buracos leves ou severos, bueiros leves ou severos, rachaduras transversais ou longitudinais, cavidades leves ou severas, faixas de desaceleração, lombadas, travessia ferroviária leve ou severa e estradas de paralelepípedo. Em M. e Gopi (2017) um *Gaussian Model* identificou buracos e lombadas. Wang et al. (2018) aplicou o *Mahalanobis Taguchi System* para identificar a tampa de bueiro, buracos e redutores de velocidade. Wickramarathne, Garg e Bauer (2018) através de *Relief Algorithm* detectou buracos, solavancos e rachaduras.

Com relação à propriocepção, Wu et al. (2013) reconheceu os eventos de direção como direção normal, aceleração, desaceleração, mudança para as pistas da esquerda ou direita, di-

reção em zigue-zague e aproximação do carro na frente através de *Hidden Markov Models*. Através de um sistema *Fuzzy Logic*, também foi estabelecido um indicador de nível de perigo em relação à condução. Choudhary e Ingole (2014) também usou *Fuzzy Logic* para identificar comportamento de direção perigoso. Arroyo, Bergasa e Romera (2016) usou um classificador *Adaptive Fuzzy* para identificar aceleração e frenagem. Eftekhari e Ghatee (2018) aplicou *Adaptive Neuro Fuzzy Inference System* para classificar a direção em segura, semi-agressiva ou agressiva. Bose et al. (2018) detectou frenagem e aceleração empregando um classificador *Random Forest*. Vlahogianni e Barmounakis (2017) usou *Rough Set Theory* para identificar aceleração, curvas à esquerda e à direita. Wu et al. (2018) aplicou *Naïve Bayes* para classificar os eventos de direção em frenagem repentina, mudança casual de faixa, curva rápida, retorno rápido e estacionamento prolongado. Mahboob et al. (2017) usou um SVM para identificar curva à esquerda ou à direita, curva em U (meia-volta), frenagem abrupta ou suave. Finalmente, Saleh, Hossny e Nahavandi (2017) aplicou uma LSTM-CNN para classificar o comportamento de direção em direção normal, agressiva ou motorista sonolento.

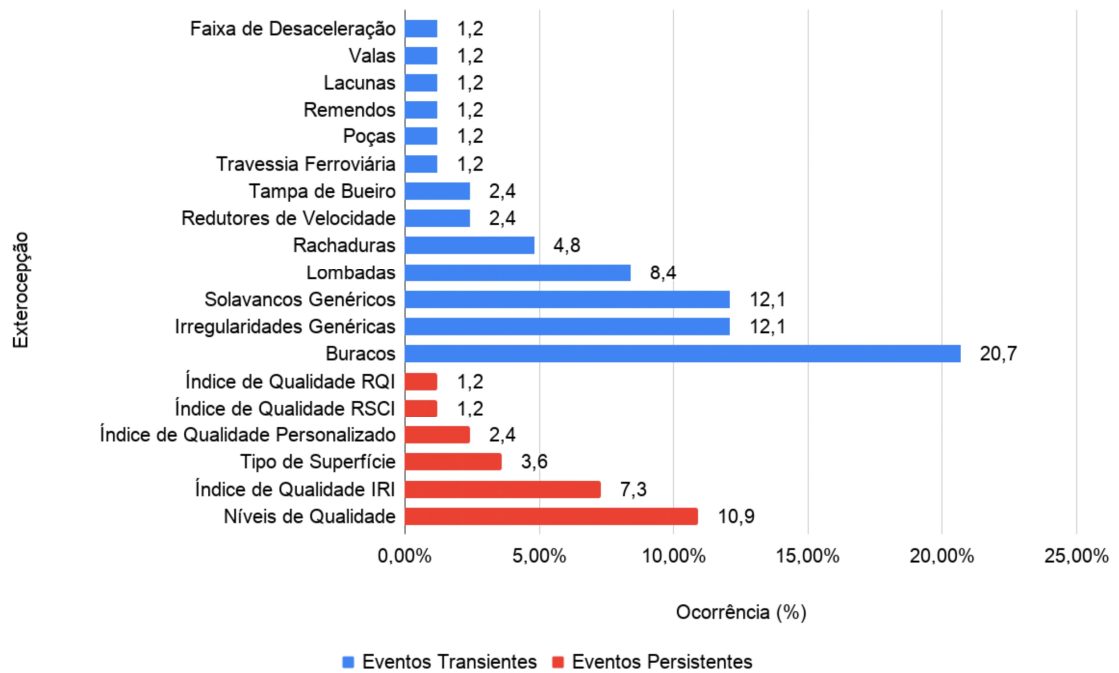
#### 4.3.6 Padrões de Percepção Veicular

Vários tipos de dados de percepção foram identificados pelos estudos analisados nesta revisão. Além da classificação entre a exterocepção e a propriocepção, as percepções também foram classificadas quanto ao momento de detecção, sendo eventos transientes aqueles detectados momentaneamente, e eventos persistentes aqueles que ocorrem a todo o momento. Sendo assim, na exterocepção, são considerados eventos transientes obstáculos na estrada e anomalias de superfície, e eventos persistentes a identificação do tipo de pavimento, os índices de irregularidade e os níveis de qualidade. Na propriocepção, eventos transientes são eventos de condução e eventos persistentes são relacionados ao perfil do comportamento de condução. As Figuras 20 e 21 detalham as percepções encontradas.

#### 4.3.7 Áreas de Aplicação

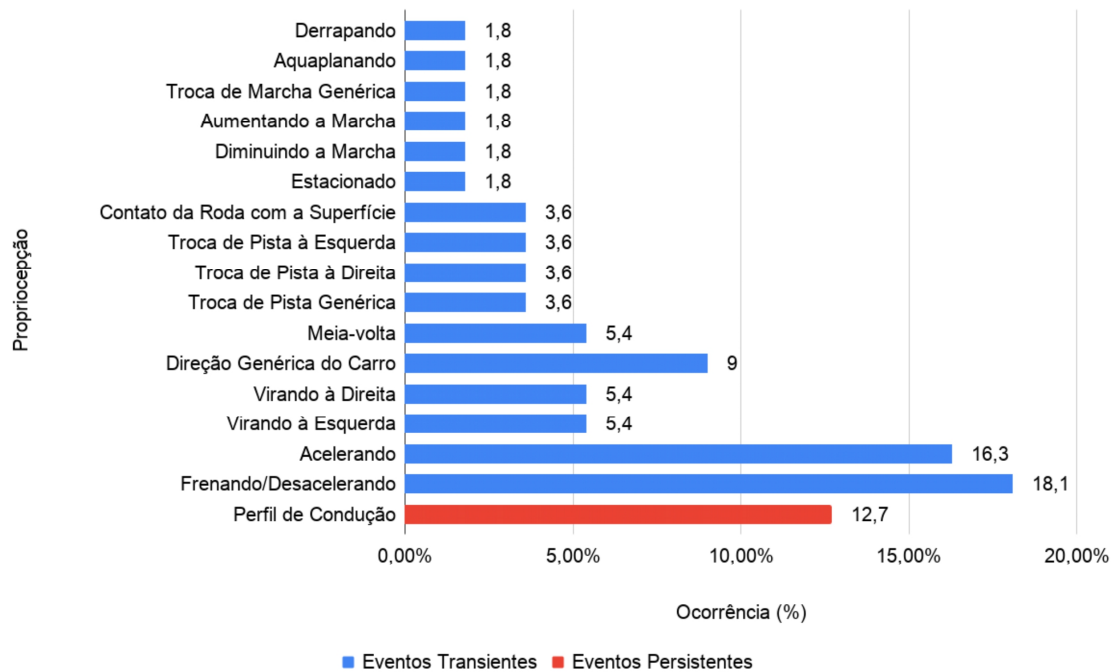
Alguns dos estudos analisados nesta revisão utilizaram as percepções veiculares desenvolvidas em uma aplicação final. Os trabalhos que utilizam sensores embarcados em dispositivos móveis, devido a produzirem dados com menor confiabilidade, geralmente são aplicados em soluções de *mobile crowdsourcing* com sensoriamento oportunista (AFRIN; MAHMUD; RAZZAQUE, 2015; GHADGE; PANDEY; KALBANDE, 2015; KAUR et al., 2017; KUMAR et al., 2017; LI; GOLDBERG, 2018; LIMA et al., 2016; POOJA; HARIHARAN, 2017; RISHIWAL; KHAN, 2016; SAVERA et al., 2016; SINGH et al., 2017; STRUTU; STAMATESCU; POPESCU, 2013; VITTORIO et al., 2014). Neste tipo de aplicação, as soluções georreferenciam os eventos de percepção, enviando-os para um servidor central. Neste servidor, uma etapa de pós-processamento é aplicada, onde dados de diferentes fontes são integrados para melhorar a confiabilidade, como em Gawad, Mougy e El-Meligy (2016) e Pooja e Hariharan (2017). Poste-

Figura 20 – Exterocepções encontradas nos estudos analisados



Fonte: Desenvolvido pelo autor.

Figura 21 – Propriocepções encontradas nos estudos analisados



Fonte: Desenvolvido pelo autor.

riormente, esses dados são disponibilizados de volta aos usuários em aplicativos como o ADAS (AFRIN; MAHMUD; RAZZAQUE, 2015; AKHTAR; PANDEY; GUPTA, 2014; NALAVDE; SURVE, 2015).

No uso de sensores inerciais fixados no veículo, os dados foram geralmente utilizados localmente. Em Selmanaj, Corno e Savaresi (2014), as percepções foram integradas ao desenvolvimento de um *airbag*. Em Tomiyama e Kawamura (2016) foi desenvolvido um sistema de monitoramento das condições da pista. Pitoňák e Filipovsky (2016) buscou realizar controle de qualidade e garantia de qualidade no processo de entrega de projetos de engenharia civil. Khaledghian e Taheri (2017) utilizou do reconhecimento do tipo de superfície de pista para controlar a velocidade do veículo.

#### 4.3.8 Discussão e Conclusões

Nesta revisão, foram analisados estudos publicados nos últimos sete anos que visam produzir padrões de percepção veicular por meio de análise multivariada de sinais de sensores inerciais. Os métodos revisados produzem vários padrões de eventos transientes e persistentes relacionados a exterocepção e a propriocepção. Constituindo uma abordagem não-intrusiva e passiva aplicada a um objeto em movimento, esses métodos permitem o monitoramento da infraestrutura de transporte e de seus participantes. Dadas suas características de serem seguros, não poluentes, de fácil instalação e de baixo custo, esses métodos se mostram muito úteis quando empregados de forma ampla, com um grande número de usuários. Nesta seção, são comparados e discutidos vários aspectos dos métodos desenvolvidos nos artigos analisados. A discussão está estruturada de acordo com as etapas revisadas, sendo elas a coleta de dados, pré-processamento e processamento. Em seguida, são discutidos aspectos que envolvem todo o processo de reconhecimento e classificação dos padrões de percepção veicular, bem como sua aplicação nos mais diversos domínios.

Na primeira etapa, a coleta de dados foi realizada em contextos muito diferentes, com cada estudo amostrando sinais por meio de um determinado tipo de sensor, em uma plataforma específica, viajando em um determinado tipo de pavimentação, modelo de veículo e diferentes modos de condução. Assim, não foi encontrada uma metodologia comum para seleção de sensores, taxa de amostragem, colocação dos sensores na estrutura do veículo e seu posicionamento em relação aos referenciais. Desta forma, várias lacunas foram identificadas. Em relação aos sensores utilizados, a maioria dos estudos utilizou apenas o acelerômetro, ao invés de aplicar o giroscópio ou ambos. Essa seleção não se deve a critérios de qualidade, mas ocorre porque o acelerômetro é mais facilmente encontrado em dispositivos móveis, nos quais se concentra a maioria dos estudos. Porém, é necessário avaliar qual sensor fornece os melhores dados para gerar as percepções, e se a combinação dos dados de ambos pode melhorar os resultados dos modelos propostos.

Em relação às configurações de amostragem, os artigos revisados não discutem a faixa de medição utilizada, não deixando claro a possibilidade de saturação dos dados do sinal. Da



mesma forma, poucos estudos justificam a escolha da taxa de amostragem, e os que o fazem definem este valor de acordo com limitações de processamento e economia de recursos da plataforma utilizada, não levando em consideração que em altas velocidades certos eventos podem não ser amostrados nas taxas estabelecidas. Assim, dada a grande quantidade de percepções veiculares que podem ser produzidas, é necessário estabelecer taxas de amostragem consistentes e não arbitrárias, permitindo a identificação das percepções de forma eficaz e eficiente, utilizando o mínimo de recursos computacionais necessários. No contexto de aplicação dos sensores inerciais em ITS é produzido valores grandes de aceleração e de taxa de rotação, de forma que o valor do erro agregado à resolução é insignificante e pode ser desprezado.

A colocação do sensor na estrutura do veículo ocorreu de várias formas, interna ou externamente à cabine. Uma vez que existem fatores de dependência relacionados às propriedades veiculares, a amostragem de dados em diferentes localizações no veículo implica em diferentes valores observados para um mesmo evento. Não foram encontrados estudos que correlacionassem sinais em diferentes partes do veículo para verificar uma possível variação na confiabilidade do reconhecimento devido a colocação do sensor. Também não houve estudos utilizando dados em mais de uma extremidade do eixo do veículo para analisar a possível melhoria nos resultados. O posicionamento dos sensores em relação aos referenciais tem sido abordado por poucos estudos, dificultando o entendimento de como os dados foram adquiridos e analisados. Finalmente, a maior lacuna nesta etapa se dá em razão da não existência de conjuntos de dados públicos desses sensores, para que se possa conduzir novas pesquisas ou comparar os métodos já propostos. Além disso, para que estes conjuntos sejam de fato úteis, é necessário que se siga toda uma metodologia de colocação na estrutura do veículo e posicionamento dos sensores de acordo com os referenciais.

Os dados foram amostrados por meio de duas plataformas, utilizando sensores embarcados em dispositivos móveis, como *smartphones* e *tablets*, ou anexados à estrutura veicular, empregando microcontroladores ou SBC. Na primeira plataforma os dados são menos confiáveis do que na segunda, uma vez que os sensores não permanecem fixos o tempo todo e não possuem uma colocação e posicionamento predefinidos. Isso implica que não há garantia de padronização do referencial de análise, ou seja, a localização dos eixos dos sensores pode variar constantemente em relação aos do veículo. Desta forma, não foram encontrados métodos aplicados localmente para garantir a qualidade dos dados amostrados em dispositivos móveis, considerando a possível interferência nestes dispositivos, principalmente humana. Alguns métodos voltados a prover maior confiabilidade dos dados foram empregados em servidores centrais que recebem padrões reconhecidos de várias fontes. No entanto, eles não se mostram úteis se as fontes primárias sempre fornecerem resultados insatisfatórios devido ao referencial incorreto. O único método que visa fornecer certa confiabilidade são as fórmulas para reorientação dos eixos baseadas nos ângulos de Euler, as quais possuem diversas limitações discutidas na etapa de pré-processamento, principalmente devido ao mapeamento dos eixos do sensor para o referencial do mundo real e não para o referencial do veículo. Nenhum dos estudos discute a comparação dos aspectos da coleta de dados nas duas plataformas apresentadas, e a possi-

bilidade de se criar um modelo de reconhecimento de percepção possível de ser utilizado em ambas. Além dos sensores inerciais, alguns estudos utilizaram fontes de dados auxiliares. O GPS e OBD-II foram usados para obter a velocidade do veículo, a qual consiste de um dos fatores de dependência. Alguns estudos também utilizaram o sensor de magnetômetro para medir o campo geomagnético ambiental, que foi empregado com os dados de aceleração para calcular os ângulos de Euler durante a reorientação dos eixos.

Uma vez coletados, todos os dados brutos foram pré-processados antes de serem passados para a técnica que reconhece e classifica os padrões de percepção. O pré-processamento foi subdividido em três etapas de acordo com sua finalidade. Na primeira etapa de pré-processamento, foram aplicados métodos de reorientação dos eixos entre os referenciais do sensor e do veículo. Diretamente relacionados aos referenciais de coleta e análise, duas técnicas foram apresentadas para esse fim: posicionamento controlado e fórmulas baseadas em ângulos de Euler. No posicionamento controlado, aplicado na etapa de coleta de dados, embora os dados sejam amostrados em um referencial correto para análise, a técnica é suscetível a erros quando utilizada em sensores não fixados no veículo, como os embarcados em *smartphones* e *tablets*. Isto se deve ao possível desalinhamento dos eixos, ocasionado pela não fixação dos sensores. Em relação aos ângulos de Euler, as três formas apresentadas para o cálculo dos ângulos possuem limitações específicas. Na primeira, apresentada em Orhan e Eren (2013), Vittorio et al. (2014), Singh et al. (2017), devido ao uso de uma orientação inicial que não é atualizada continuamente, os valores em locais com aclive ou declive do terreno podem apresentar erros. Isso se deve ao fato de que nesses locais a força de aceleração gravitacional é distribuída entre os eixos, não ficando isolada no eixo de aceleração vertical. Na segunda técnica apresentada em Li e Goldberg (2018), embora os ângulos sejam atualizados, o valor da força gravitacional não é isolada e pode levar a erros por não ser o único componente de força presente nos eixos. Na terceira técnica de Singh et al. (2018), devido à tentativa de identificar um possível estado estacionário durante a condução do veículo para defini-lo como um novo referencial inicial, isso implica que estradas de alta vibração não satisfazem as condições e não atualizam os ângulos, acumulando erros.

Todos os métodos apresentados de cálculo dos ângulos de Euler para reorientação dos eixos são baseados na localização da força de aceleração gravitacional. Sendo assim, os ângulos de rotação devem sempre ser calculados a partir de um estado estacionário, e não em relação aos dados de aceleração atuais como feito em alguns estudos. As fórmulas matemáticas e as condições de aplicação podem ser encontradas no estudo primário Astarita et al. (2012), que é referenciado e utilizado pelos artigos revisados. Ainda por serem baseadas na gravidade, as técnicas reorientam o eixo Z perpendicularmente ao solo, apontando para o céu, ao invés de ser perpendicular ao chão do veículo. Portanto, os dados são reorientados para o referencial do mundo real e não para o referencial do veículo, que apenas se sobrepõe ao eixo Z em casos de terreno plano. Essa técnica de reorientação somente é útil quando usada em dispositivos móveis com sistemas de *mobile crowdsensing*, produzindo apenas exterocepção. Nos métodos de reorientação analisados, os eixos X e Y ficam sem uma orientação bem definida.

Para a propriocepção, mesmo adicionando orientação ao eixo X e Y através do magnetômetro, o referencial do mundo real (terra) não permite identificar eventos de condução, sendo necessário estar no referencial do veículo. Convém frisar que os métodos apenas são aplicáveis a reorientação dos dados de aceleração, e não para os de taxa de rotação. Portanto, a abordagem do posicionamento controlado, embora simples, com a fixação de sensores e alinhamento dos eixos apresenta menor possibilidade de erros, permitindo reconhecer a exterocepção e a propriocepção.

Na segunda etapa de pré-processamento de sinal, foram aplicados métodos para remoção de ruídos, suavização de sinal, segmentação de dados e extração de características. Em resumo, os estudos aplicaram métodos estatísticos no domínio do tempo, e métodos como transformadas e *wavelets* no domínio da frequência. No domínio do tempo, métodos como médias simples ou móveis foram aplicadas para remover ruídos e suavizar os sinais. Métodos como o desvio padrão e o RMS foram aplicados para extrair características de alto nível que bem representassem os dados para o reconhecimento de padrões. No domínio da frequência, métodos como IIR e *Butterworth* foram utilizados para remover ruído, através de frequências específicas, como a da gravidade nos dados de aceleração. A FFT e as *wavelets* foram empregadas para extrair características tais como magnitudes das vibrações. Não encontramos uma análise comparativa dos métodos de remoção de ruído e extração de características em seus diferentes domínios, sendo uma lacuna a ser explorada. Em relação ao domínio de processamento, a maioria dos métodos foi aplicada no domínio do tempo, sendo que os domínios da frequência e do tempo-frequência devem ser melhor explorados, para se verificar a possibilidade de produzir dados mais relevantes para as técnicas de classificação. A segmentação dos dados foi realizada de quatro formas: em função do tempo, número de amostras, distância ou velocidade. Em geral, o uso da janela em função da distância é mais adequado. Não encontramos nos estudos analisados um valor ideal para o tamanho da janela de dados. Em pesquisas futuras, o impacto do valor deste parâmetro deve ser avaliado.

Na terceira etapa do pré-processamento, foram aplicados os métodos de georreferenciamento dos dados. Uma vez que a taxa de amostragem de dados de sensores inerciais é muito maior do que a de dados de localização (1 Hz), são necessárias técnicas para estimar a localização (posição mais granulada) e a velocidade, antes da próxima amostra de GPS. Para fazer isso, apenas Li e Goldberg (2018) utilizou um tratamento, aplicando interpolação linear com os dados de aceleração para obter dados de localização com mais frequência. Porém, o mesmo não foi feito com a velocidade. O desenvolvimento de um método para esse fim é de extrema importância, uma vez que a velocidade constitui um fator de dependência e deve ser o mais próximo possível do real. Da mesma forma, estimar pontos de localização ajuda a mitigar pontos cegos, reduzir o erro de localização, entre outros problemas. Não identificamos nenhuma desvantagem em usar este método, exceto o aumento do processamento do modelo.

Na etapa de processamento, várias técnicas foram propostas para reconhecer e classificar os padrões nos dados pré-processados dos sensores. Embora alguns trabalhos apresentem técnicas com abordagens mais elaboradas, como métodos de alinhamento de séries temporais,

aprendizado de máquina e modelos com conhecimento especializado, quase metade deles utiliza de técnicas extremamente simples, como limiares e funções lineares. As demais técnicas empregadas são geralmente clássicas, sendo necessário testar e avaliar técnicas mais recentes. Dentre as abordagens mais recentes, o *Deep Learning* é um dos que apresenta o maior potencial para melhorar os resultados de reconhecimento e classificação das percepções veiculares, o qual nos últimos anos fez grandes avanços nos mais variados domínios (LECUN; BENGIO; HINTON, 2015). Uma das grandes vantagens das técnicas de *Deep Learning* está em adicionar a segunda etapa de pré-processamento ao modelo de treinamento. Dessa forma, o melhor método para remover ruído e extrair recursos de alto nível dos dados brutos é definido por meio do treinamento. Nos estudos analisados, essa abordagem foi encontrada em apenas uma pesquisa, a de Saleh, Hossny e Nahavandi (2017), a qual se limita a um tipo específico de percepção. Portanto, é uma abordagem que deve ser mais explorada nesta área de pesquisa.

Em relação às variáveis de entrada utilizadas pelas técnicas, a maioria dos estudos utilizou apenas dados do eixo em que as percepções são mais evidentes. Assim, analisando no referencial do veículo, os padrões de exterocepção são mais evidentes na aceleração do eixo Z e nas taxas de rotação dos eixos X e Y. Já os padrões de propriocepção são mais evidentes nos dados de aceleração dos eixos X e Y e na taxa de rotação do eixo Z. Contudo, embora não visualmente evidentes, as percepções também produzem dados significativos nos demais eixos, conforme discutido em M. e Gopi (2017). Portanto, é importante considerar todos os eixos dos sensores inerciais no reconhecimento de percepção, bem como a velocidade do veículo. Além disso, os estudos geralmente se concentram no reconhecimento de uma percepção específica, não trabalhando a correlação e integração de diferentes tipos de percepção, bem como na integração entre a exterocepção e a propriocepção. Essa integração se deve ao fato de que várias percepções são dependentes ou correlacionadas entre si. Um exemplo de dependência ocorre no reconhecimento do tipo de composição da superfície da estrada que depende da ausência de derrapagem ou aquaplanagem. Quanto à correlação, em geral a percepção de desaceleração por frenagem está relacionada a algum obstáculo, como buraco ou lombada. Adicionar essas percepções como variáveis de entrada para o reconhecimento de outras pode melhorar os modelos.

Os resultados dos métodos propostos não foram comparados nesta revisão por vários motivos. Em primeiro lugar, a maioria dos estudos não tem resultados medidos ou métricas semelhantes para comparação. A metodologia aplicada também não é clara em muitos deles, carecendo de muitas informações relevantes. Além disso, cada trabalho foca em algum tipo específico de percepção, não permitindo a comparação direta devido aos diferentes propósitos dos métodos. Por fim, a maior limitação para a não comparação se deve ao fato de os dados serem amostrados em contextos significativamente distintos, conforme detalhado anteriormente, fazendo com que as comparações tornem-se subjetivas, principalmente pela existência de fatores de dependência. Da mesma forma, a ausência de conjuntos de dados públicos impossibilitou a realização de testes de avaliação dos métodos. Assim, esta revisão teve como foco a análise de aspectos dos métodos e técnicas aplicados em todas as etapas da coleta e análise de dados.

Através da análise dos estudos, foi observado um enfoque maior na simples e imediata aplicabilidade dos resultados do que em fornecer dados mais precisos e acurados com uma certa adaptabilidade. Isso mostra-se mais evidente em aplicações de sensores embarcados em dispositivos móveis. Sendo assim, a maioria dos trabalhos foca na aplicação das percepções produzidas, e a adaptabilidade das soluções apresentadas é pequena, sendo o principal obstáculo para sua ampla utilização em cenários do mundo real. Em relação aos fatores de dependência, a adaptabilidade raramente é mencionada nos artigos. Alguns estudos adicionaram em seus modelos o fator de velocidade ou o modelo *Quarter Car* (QC) (TOMIYAMA; KAWAMURA, 2016), porém nenhum deles trata de todos os fatores de dependência para produzir diferentes tipos de percepção. Além disso, nenhum dos estudos experimenta o modelo *Half-Car* (HC), que pode ser mais interessante em aplicações de *mobile crowdsensing* que o QC. Singh et al. (2017) argumentam que não é possível cobrir todas as condições por meio de aprendizado de máquina e modelos de limiares existentes. No entanto, a adaptabilidade deve ser investigada do ponto de vista das três etapas apresentadas, especialmente integrando os dados com as novas técnicas robustas de aprendizado de máquina. Outro ponto a ser explorado no sentido de adaptabilidade e ainda não observado nos estudos é a integração dos dados de percepção, conforme mencionado anteriormente. Por exemplo, eventos persistentes que descrevem as condições gerais da pista de um determinado segmento de estrada podem ser considerados como fatores de dependência: o tipo de composição e o estado de conservação da superfície implicam em padrões diferentes para o mesmo evento transiente. Assim, um buraco detectado no pavimento asfáltico pode ter um padrão diferente de buracos em outros tipos de pavimento. Isso também deve ser levado em consideração por abordagens adaptativas.

Em relação ao tempo de detecção, os estudos produziram as percepções de forma *offline*, *online* e em tempo real. No entanto, a maioria deles confunde os termos *online* e tempo real, havendo necessidade de em novos estudos analisar restrições temporais de resposta dos modelos. Em relação à aplicação, nos estudos analisados, os dados de percepção veiculares foram utilizados principalmente em ADAS. Porém, por consistir em um segmento dentro de ITS, novas aplicações podem se beneficiar dessas identificações como caixa preta veicular para seguradoras, sistemas de informações situacionais de planejamento de manutenção de estradas ou aplicados a agentes inteligentes para navegação autônoma. Além disso, embora uma ampla gama de percepções possa ser observada nos estudos, identificamos situações em que é possível gerar novas percepções a partir dos sensores inerciais, tais como a identificação de relevo.

#### 4.4 ANÁLISE DE PERCEPÇÕES VEICULARES ESPECÍFICAS

Nesta seção, embasado no estado da arte levantado pela revisão da literatura, é discorrida uma análise comparativa dos aspectos mais relevantes dos trabalhos relacionados às três percepções abordadas nesta pesquisa. Para isso, a RSL foi atualizada, compreendendo trabalhos publicados até 2021. Nas seções subsequentes são analisados, respectivamente, os trabalhos de classificação de tipo de superfície de pista, de qualidade de superfície e de reconhecimento de

lombadas.

#### 4.4.1 Classificação do Tipo de Superfície de Pista

Baseando-se nas propriedades de dependência apresentadas, analisamos os modelos propostos em pesquisas relacionadas a fim de verificar quais propriedades foram adotadas. Os estudos para classificação de tipo de superfície com sensoriamento inercial encontrados foram conduzidos em dois tipos de veículos: *wheeled ground robots* (KHALEGHIAN; TAHERI, 2017; SEBASTIAN; BEN-TZVI, 2019; TOLENTINO-RABELO; MUÑOZ, 2016) e carros (SOUZA, 2018; WANG et al., 2018; WANG et al., 2017). Uma vez que a estrutura entre os dois tipos de veículos é muito diferente, e existe a propriedade de dependência veicular, os trabalhos de *wheeled ground robots* não foram considerados. Dentre os estudos que aplicam os sensores em carros, Souza (2018) utilizou sensores embarcados em *smartphones* e Wang et al. (2018), Wang et al. (2017) fixaram no veículo.

Em Souza (2018) o *smartphone* com sensores foi fixado dentro do veículo usando um suporte de sucção flexível perto do painel. O modelo desenvolvido utilizou de dados do acelerômetro (100 Hz) e velocidade do GPS, obtendo os melhores resultados com o modelo de LCSS combinado com CID. Como resultado, a superfície de pista é classificada entre asfalto (98,28%), paralelepípedo (84,41%) e terra (78,64%), obtendo na média 87,77% de precisão. Em Wang et al. (2018) e Wang et al. (2017) foi utilizado dados do acelerômetro montado na suspensão veicular e velocidade do GPS. No pré-processamento foi empregado o modelo matemático QC, de forma a considerar a influência da suspensão veicular, e FFT para extração de características. Estes dados foram treinados em uma SVM, classificando a pista em asfalto (17,6%), concreto (99,6%), grama (74,9%) e terra (85,3%), com média de precisão de 69,4%.

Analisando os estudos, observamos que são limitados ao uso de dados de aceleração, velocidade e modelo QC. Sendo assim, não são exploradas todas as propriedades de dependência e não é analisado o comportamento do modelo em variações contextuais. O ambiente de coleta não é detalhado adequadamente, de forma não ser possível verificar condições de conservação dos pavimentos ou presença de irregularidades e obstáculos. Também não há especificação dos veículos utilizados e seus condutores, e se foram utilizados mais de um. Por fim, os resultados obtidos apresentam baixos valores das métricas de avaliação, mesmo em cenários controlados.

#### 4.4.2 Classificação da Qualidade de Superfície de Pista

Baseado no estado da arte e nos aspectos relevantes levantados, como os fatores de dependência, analisamos os modelos propostos em pesquisas relacionadas para verificar quais propriedades foram levadas em consideração. Para classificação da qualidade da superfície de pista, os estudos se concentraram na criação de índices de irregularidade e níveis de qualidade. Os trabalhos correlatos empregaram sensores em *smartphones* (DOUANGPHACHANH;

ONEYAMA, 2014b; LIMA et al., 2016; BRUNAUER; REHRL, 2016; ZHAO et al., 2016; ALLOUCH et al., 2017; SOUZA; GIUSTI; BATISTA, 2018; LI; GOLDBERG, 2018; NUNES; MOTA, 2019; TIWARI; BHANDARI; RAMAN, 2020; BADUROWICZ; MONTUSIEWICZ; KARCZMAREK, 2020; ABDELRAHEEM; EL-MELEGY, 2020) e fixados em veículos (CHEN et al., 2013; CHEN et al., 2016; PITOŇÁK; FILIPOVSKY, 2016; PRAPULLA; RAO; HERUR, 2017; PONT; PROVOST; KUENZEL, 2017; LEI; MOHAMED; CLAUDEL, 2018; HASSAN et al., 2019; ABDELRAHEEM; EL-MELEGY, 2020; MEOCCI; BRANZI; SANGIOVANNI, 2021).

Em Nunes e Mota (2019) foi apresentado um *framework* para classificar a qualidade de uma estrada entre boa e ruim, experimentado em vias urbanas. Para isto, um aplicativo coleta localização e velocidade através do GPS, além de dados do acelerômetro, ambos embarcados em *smartphones*. Junto com os dados, o dispositivo coleta a opinião do usuário sobre a via, para utilizar como rótulo das classes de dados. Os dados coletados são enviados para um servidor na nuvem, que pré-processa os dados, empregando com modelos de *Random Forest*, KNN, J48 e SVM, os quais obtiveram acurácia respectiva de 90,64%, 86,04%, 75,67% e 74,97%. Em Douangphachanh e Oneyama (2014b) foram realizados experimentos para analisar a relação entre as frequências dos dados de aceleração e os níveis de irregularidade da via. Neste estudo, foram coletados dados de acelerômetro (100 Hz) e GPS embarcados em *smartphones*. Os dispositivos foram colocados em diferentes locais, como bolso da camisa do motorista e próximo ao câmbio de marchas, e em diferentes veículos. Com os dados coletados, foi realizado uma análise no domínio da frequência através da aplicação de FFT, para verificar a relação entre a magnitude das frequências e as classes de dados de qualidade da via: boa, regular, ruim e péssima.

Em Souza, Giusti e Batista (2018) foi proposto um sistema para monitorar as condições do pavimento. Para isto, utilizaram um *smartphone* com sensores fixado dentro do veículo usando um suporte de sucção flexível perto do painel. O modelo desenvolvido utilizou de dados do acelerômetro (100 Hz) e velocidade do GPS. Através dos dados coletados, foram extraídas características no domínio do tempo, com métodos estatísticos, e na frequência, com FFT, aplicadas em SVM, *Random Forests*, *Naive Bayes*, *Decision Trees*, e KNN. O modelo de SVM apresentou os melhores resultados, classificando com 92,39% de acurácia os segmentos entre bom, regular, ruim, péssimo e trecho com obstáculos. As classes de dados foram anotadas a partir da percepção de pessoas. Tiwari, Bhandari e Raman (2020) utilizaram de acelerômetros em *smartphones* para classificar a via entre boa, mediana e ruim. O GT foi feito de forma humana, através da observação das gravações de câmeras. Os dados foram coletados em ambiente urbano, somente em vias pavimentadas, sumariamente asfalto. Através dos dados de aceleração, foram extraídas características por meio de métodos estatísticos. Estas características foram aplicadas em uma SVM, uma *Feed-forward Neural Network* e uma CNN, as quais obtiveram respectivamente acurácia de 65,6%, 65,7%, e 98,5%.

Em Badurowicz, Montusiewicz e Karczmarek (2020) foi desenvolvida uma heurística de limiares junto com um sistema *Fuzzy Logic* para classificar a qualidade da via entre boa, medíocre e pobre. O estudo utiliza dados de um acelerômetro em *smartphone*. Leizerovych et

al. (2020) desenvolveram uma aplicação para análise da condição da via. Para isto foi utilizado de acelerômetro, giroscópio e GPS em *smartphones*. Os dados coletados em vias de asfalto foram empregados em uma CNN para classificar a pista entre lista, satisfatória e insatisfatória. A acurácia do estudo foi de 84,81%. Allouch et al. (2017) apresentaram modelos para classificar a qualidade da via. Para isto, foi utilizado de acelerômetro, giroscópio e GPS em *smartphones*. No pré-processamento foi utilizado FFT para extração de características. Para classificar as vias entre normal e esburacada, as características foram aplicadas em *Decision Tree*, *SVM* e *Naive Bayes*, as quais obtiveram acurácia de 98,50%, 95,25% e 96,90%, respectivamente.

Em Lima et al. (2016) foi desenvolvida uma aplicação para estimar a qualidade da via, assim como problemas transientes no asfalto. Para isto, utilizaram de acelerômetro e GPS em *smartphones*. Os dados de aceleração e velocidade foram empregados em uma heurística de limiares para classificar a via em boa, normal, ruim e péssima. Brunauer e Rehr (2016) desenvolveram uma aplicação para calcular o RCSI. Para isso utilizaram de acelerômetro e GPS em *smartphones*. Para calcular o RCSI, foi empregado uma função linear simples que mapeia a quantidade de eventos transientes detectados para cinco níveis de qualidade (1-5). AbdelRaheem e El-Melegy (2020) utilizaram GPS e acelerômetros fixados no veículo e em *smartphones* para calcular o IRI. Os sensores foram utilizados em um único carro, em velocidade entre 30km/h e 40km/h. O estudo utilizou apenas a aceleração vertical na fórmula simplificada de cálculo IRI.

Em Li e Goldberg (2018) foi apresentado um aplicativo de *mobile crowdsensing* para analisar irregularidades na superfície da estrada. Para isso, utilizaram de um acelerômetro e GPS em *smartphones*, fixados na cabine do veículo por meio de suporte. No acelerômetro, os sinais foram processados para reorientar os dados, com fórmulas baseadas nos ângulos de Euler. Posteriormente, usando os filtros passa-alta e passa-baixa, o componente de gravidade foi removido. Em segmentos de 50 metros foi calculado o índice IRI-proxy, através da normalização do RMS de aceleração vertical pela velocidade. Zhao et al. (2016) desenvolveram uma aplicação para calcular o IRI. Para isso utilizaram de acelerômetro e GPS em *smartphones*, utilizado em três veículos e quatro diferentes locais de coleta, todos acima da suspensão. A velocidade empregada ficou entre 40km/h e 60km/h. Para calcular o IRI foi empregado os dados de aceleração vertical no modelo HC e fórmulas do IRI. Os parâmetros do modelo HC foram definidos por um algoritmo genético.

Em Chen et al. (2013) e Chen et al. (2016) foi desenvolvido um sistema de *crowdsourcing* para monitoramento da superfície de pista. O estudo utiliza um conjunto de *hardware* fixado no veículo, composto por um acelerômetro (100 Hz), um GPS e um microcontrolador. O desvio padrão dos dados de aceleração são empregados em uma função linear simples para estabelecer o IRI. Posteriormente, outra função linear mapeia o IRI para o RQI, e deste para quatro níveis de qualidade: excelente, boa, qualificada e desqualificada. Lei, Mohamed e Claudel (2018) apresentaram uma plataforma para monitoramento de superfície da via. Neste estudo foi utilizada apenas a aceleração vertical amostrada em acelerômetros conectados a um microcontrolador. A plataforma foi fixada na entrada USB do veículo. Os rótulos das classes de dados



foram estabelecidos com base na opinião de pessoas, classificando a qualidade da via em um índice *Present Serviceability Rating* (PSR) de 0-5. Os dados de aceleração foram aplicados em uma FFT, onde foi analisado a relação entre as diferentes frequências e os níveis de qualidade.

Em Meocci, Branzi e Sangiovanni (2021) desenvolveram uma abordagem para estabelecer dois índices de irregularidade da via, assim como conceitos de qualidade relacionados. Para isto, o estudo utilizou de sensores inerciais (100 Hz) e GPS embarcados em caixas pretas veiculares. Os dados foram coletados em vias asfaltadas de área urbana, em velocidade de 30 km/h até 50 km/h. O método utilizou da aceleração vertical e velocidade coletadas em métodos estatísticos tais como média móvel e desvio padrão para estabelecer dois índices de irregularidade: *Local Pave Box index* (LPBi) e *Global Pave Box index* (GPBi). Através de limiares, estes índices foram mapeados para níveis os qualidade bom, regular, ruim e trecho precisando de reparos. Pont, Provost e Kuenzel (2017) utilizaram de acelerômetros e GPS no veículo para avaliar o conforto de condução com base na qualidade da via. Através de uma função simples que tem como entrada a aceleração e velocidade, é estimado o RQI. Este índice é comparado com limiares para classificar a via entre crítica e não crítica.

Em Hassan et al. (2019) foi desenvolvido um método para classificação da qualidade de pista baseado na quantidade de obstáculos e irregularidades. Para isso, um acelerômetro e GPS foram instalados em veículos, e seus dados foram processados para identificar eventos como lombadas e buracos. De acordo com a quantidade de eventos encontrados, a via é classificada entre boa, regular, ruim e terrível. Prapulla, Rao e Herur (2017) utilizaram um acelerômetro e GPS em um Arduíno. Nos dados de aceleração é empregado desvio padrão, valor utilizado para segmentar a qualidade da via. Pitoňák e Filipovsky (2016) utilizou de um giroscópio instalado no veículo para cálculo do índice IRI. Para isto, foi utilizado o valor *pitch* em uma função linear. Posteriormente, o resultado da função é aplicado em outra função que mapeia para o IRI, através de correção.

Através da análise dos trabalhos relacionados, é possível observar que nenhum deles busca considerar todos os fatores de dependência. Alguns dos estudos consideram diferentes velocidades ou diferentes veículos, embora nenhum dos estudos considere todos os fatores em conjunto. Em relação aos ambientes, os estudos na sua grande maioria realizam a identificação de qualidade de pavimentos asfálticos, não analisando outros pavimentos ou a vias sem pavimentação. Os níveis de qualidade, em muitos casos, são simples mapeamentos de um índice personalizado de irregularidade para conceitos, não se mostrando confiáveis uma vez que os modelos que geram os índices são extremamente simplistas. Em relação à quantificação de irregularidade pelos mais variados índices, nota-se que estes indicadores são fortemente dependentes dos conjuntos de dados utilizados, isto porque os fatores de dependência não são levados em consideração, e sumariamente os índices consideram apenas a aceleração vertical no cálculo.

Os índices em geral constituem de funções lineares simples de um componente apenas. No caso do IRI, embora suas fórmulas sejam robustas e considerem toda a mecânica veicular envolvida, os estudos utilizam das fórmulas reduzidas pela aplicação dos parâmetros *Golden*

*Car*, os quais padronizam diversos aspectos, tais como velocidade constante em 80 km/h. Estes parâmetros não são adequados para aplicação real justamente por se limitarem a um cenário específico, um determinado sistema de amortecimento, determinada velocidade, etc. Em aplicações do mundo real, estes valores possuem grande variação. Em relação a rotulação dos dados, praticamente todos utilizam de anotação manual de GT humano, tornando extremamente subjetivo o que pode ser classificado como bom ou ruim. Mostra-se necessário desenvolver um GT anotado por máquina para não haver diferentes percepções de qualidade em relação a um mesmo segmento de via. Em relação aos dados utilizados, muitos utilizam apenas um eixo do sensor para fazer análise. Diversos estudos relacionados se mostram prova de conceito, faltando detalhes do ambiente de coleta, colocações dos sensores, e métricas de resultados insuficientes para análise aprofundada.

#### 4.4.3 Reconhecimento de Lombadas

Através dos estudos analisados, é possível observar que a grande maioria possui foco na identificação de eventos em uma forma genérica, em classes de dados tais como *obstáculo*, *anomalia* ou *irregularidade*. Considerando como exemplo buracos e lombadas, agrupá-los em um único identificador torna a aplicação de ITS menos inteligente, especialmente se considerarmos que um buraco é uma irregularidade que deve ser corrigida e deixar de existir, enquanto que a lombada é um obstáculo que sempre estará lá. Desta forma, a especificação da informação situacional durante a detecção se mostra importante, e nesta seção são analisados somente os estudos que realizam a identificação de lombadas como uma de suas classes de dados.

Para identificação de lombadas através de sinais de sensores inerciais, a coleta de dados ocorreu com a utilização destes dispositivos em duas formas: fixados no veículo (SABIR; MEMON; SHAIKH, 2019; CELAYA-PADILLA et al., 2018) ou embarcados em *smartphones* (FOUAD et al., 2014; ALAM et al., 2020; RISHIWAL; KHAN, 2016; WANG et al., 2018; VERMA et al., 2016; SOUZA, 2018; EDWAN; SARSOUR; ALATRASH, 2019; SAVERA et al., 2016; IDRIS; BOUKAR; ADESHINA, 2019; DEY et al., 2019; ALJAAFREH et al., 2017; ARAGÓN et al., 2016). No trabalho de Rishiwal e Khan (2016) foi desenvolvida uma aplicação para detecção de buracos e lombadas. Os dados foram coletados através de acelerômetro e GPS em um *smartphone*. Para reconhecer os padrões foram utilizados limiares condicionados por intervalos de velocidade. A acurácia da abordagem proposta foi de 93,75%. Em Verma et al. (2016) foi desenvolvido uma aplicação para estimar o tempo de viagem utilizando localização de lombadas, curvas e paradas obrigatórias. Como fonte de dados, o estudo utilizou acelerômetro, giroscópio, bússola e GPS. Para detectar lombadas, foram utilizados limiares com os valores de aceleração no eixo Z, obtendo acurácia de 87,5%. Edwan, Sarsour e Alatrash (2019) desenvolveram uma aplicação para dispositivos móveis que detecta e avisa lombadas na via. Foram utilizados acelerômetro e GPS em *smartphones*. As lombadas foram reconhecidas através de limiares.

Savera et al. (2016) desenvolveram um aplicativo para detectar e alertar lombadas e bu-

racos na pista. Para isto, foram utilizados sensores acelerômetro e GPS em *smartphones*. Os dados foram coletados de diferentes dispositivos e em diferentes veículos. Posteriormente, os dados foram aplicados em uma SVM. Na detecção de lombadas o estudo obteve 86,34% de acurácia, 90,80% de precisão, 94,61% de *recall*, 92,67% de *f1-score*. Idris, Boukar e Adeshina (2019) desenvolveram uma aplicação para reconhecer buracos e lombadas. O estudo utilizou de dados de um acelerômetro embarcado em *smartphone*. Os dados foram coletados em um mesmo segmento de via com diferentes veículos. Após coletados, os dados foram pré-processados em segmentos de desvio padrão aplicados em uma SVM. Dey et al. (2019) desenvolveram uma aplicação para monitorar condições da via, dentre as quais a identificação de lombadas. Para isto, utilizaram de acelerômetro e magnetômetro embarcados em *smartphones* colocados em diferentes locais, como painel, suporte para copo e dentro do bolso. As análises foram feitas utilizando diversos algoritmos de classificação, como *Random Forest*, *Random Tree* e SVM. A classificação foi feita após a seleção de características usando os algoritmos *GreedyStepwise*, *Ranker* e *BestFirst*, considerando o valor mínimo, máximo, mediana e desvio padrão como recursos estatísticos. O estudo obteve 92% de precisão para detectar as condições da superfície da estrada.

Aragón et al. (2016) desenvolveram um modelo para classificar diferentes tipos de anomalias presentes em estradas, como buracos e lombadas. Para isto, foram utilizados dados de acelerômetros em *smartphones*. Os dados foram inicialmente segmentados em janelas de dados rotuladas em anomalia e não-anomalia através de uma SVM. Em seguida, os segmentos foram aplicados em uma representação *Bag of Words* e *Random Forest* para classificar os segmentos, obtendo 71% de acurácia. Alam et al. (2020) propuseram um sistema que detecta três eventos de estrada: lombadas, buracos e remendos. A coleta de dados empregou acelerômetro, giroscópio, GPS e magnetômetro em *smartphones*, utilizando cinco veículos de duas rodas, sete de três rodas e três de quatro rodas com seis diferentes tipos de *smartphones* em diferentes locais e posicionamentos. O sistema proposto funciona em duas fases. A primeira fase é executada em um *smartphone* e identifica as assinaturas candidatas para anomalias de estrada usando algoritmos de auto-orientação e autoajuste de limiar para torná-lo quase invariável quanto à colocação, posicionamento, veículo e tipo de *smartphone*. A segunda fase funciona em um servidor e usa um classificador baseado em *Decision Tree* para reduzir as ocorrências de falso-negativo e falso-positivo causadas pelo impacto de diferentes manobras de direção, suspensões de veículos, etc. Depois de totalmente treinado, a precisão média para lombadas foi de 98% em uma estrada lisa e 92% em estradas irregulares. A identificação de lombadas obteve 93% de precisão, 93% de *recall*, e 93% de *f1-score*.

No estudo de Wang et al. (2018) foi desenvolvida uma aplicação de coleta e processamento de dados para reconhecer condições da superfície de pista. Os dados foram coletados através de GPS, acelerômetros e giroscópios embarcados em *smartphones*, colocado dentro do veículo. Os dados foram pré-processados com transformada *wavelet*. Em seguida, estes dados foram utilizados em um modelo baseado em *Mahalanobis Taguchi System* (MTS) para detectar tampa de bueiro, buraco e lombada. Em Souza (2018) foi desenvolvido um aplicativo

para avaliação de pavimento asfáltico, reconhecendo tipos de superfície de pista, condição de conservação, obstáculos e lombadas. Para isto, foi utilizado de acelerômetro embarcado em *smartphones*, o qual foi instalado dentro do veículo usando um suporte de sucção flexível perto do painel. Para reconhecimento dos padrões, o estudo propôs a combinação de alguma medida de distância clássica para séries temporais como DTW e LCSS com CID. Através de uma avaliação experimental realizada em três conjuntos de dados que representam diferentes cenários de classificação de pavimento asfáltico, o estudo reconheceu lombadas com 81,13% de acurácia, 87% de precisão, 93% de *recall* e 90% de *f1-score*.

Aljaafreh et al. (2017) propuseram um modelo para detecção de lombadas através de um modelo de inferência de *Fuzzy Logic*. O modelo utiliza a velocidade do veículo e a aceleração obtida pelo acelerômetro do *smartphone*. O método proposto foi testado e avaliado em diferentes níveis de velocidade. Fouad et al. (2014) desenvolveram uma aplicação para detectar a presença de lombadas. Os dados foram coletados de acelerômetro, giroscópio e GPS embarcados em *smartphones*, colocados no painel do veículo. As lombadas foram detectadas através de um modelo de *Rough Mereology Theory*. O estudo obteve para reconhecimento de lombadas com acurácia de 75%, precisão de 75,4% e *recall* 16,5%.

Sabir, Memon e Shaikh (2019) desenvolveram um sistema de monitoramento de estradas usando abordagem de *crowdsourcing* para localizar buracos e lombadas. Neste estudo, foram utilizados os sensores acelerômetro e GPS embarcados em *smartphones*, e outro conjunto conectado a um Arduíno e fixado no veículo. O Arduíno e o *smartphone* foram colocados no painel, no banco da frente e no para-brisa. Os dados foram coletados em um veículo dirigido em diferentes velocidades. Através de método baseado em limiares, foram detectados buracos e lombadas e classificados quanto altura, largura e severidade. O estudo obteve de verdadeiros positivos de 90% para lombadas e 85% para buracos. Em Celaya-Padilla et al. (2018) desenvolveram um método para a detecção de lombada. Este método faz uso de um Raspberry Pi com giroscópio, um acelerômetro e um sensor GPS montado no meio da frente do carro. Os dados coletados foram empregados em uma estratégia de validação cruzada, onde um algoritmo genético foi usado para encontrar um modelo logístico que detecta anomalias na estrada. O modelo proposto teve uma acurácia de 97,14%, 93,55% de *recall*, 93,55 de precisão.

Através da análise dos trabalhos relacionados, é possível observar que nenhum deles busca considerar todos os fatores de dependência. Alguns dos estudos consideram diferentes velocidades no modelo de reconhecimento (RISHIWAL; KHAN, 2016; ALJAAFREH et al., 2017; SABIR; MEMON; SHAIKH, 2019), ou diferentes veículos (SAVERA et al., 2016; IDRIS; BOUKAR; ADESHINA, 2019; ALAM et al., 2020), ou diferentes ambientes (CELAYA-PADILLA et al., 2018), embora nenhum dos estudos considere todos os fatores em conjunto. Em relação aos ambientes, é importante notar que os estudos realizam o reconhecimento de lombadas em asfalto, onde a identificação é feita mais facilmente uma vez que a superfície é mais suave, menos irregular. Conforme Alam et al. (2020), a precisão de detecção de lombadas cai significativamente (43-80%) em estradas irregulares. Sendo assim, é necessário desenvolver um modelo que identifique bem lombadas em pavimentos irregulares, como

em segmentos de paralelepípedo. Dentre todos os trabalhos analisados, apenas dois utilizam de sensores fixados no veículo e nenhum deles analisa dados provindos de colocações abaixo da suspensão veicular. Diversos estudos relacionados se mostram prova de conceito, faltando detalhes do ambiente de coleta, colocações dos sensores, e métricas de resultados insuficientes para análise aprofundada. Conforme observado nesta análise e reafirmado por Celaya-Padilla et al. (2018), a maioria das abordagens atuais de detecção de lombadas utiliza apenas limiares simples ou de uma combinação de filtros digitais para suprimir o ruído e um modelo de classificação clássico.



## 5 CONJUNTOS DE DADOS

Para realizar os experimentos deste estudo foi necessário dispor de conjuntos de dados amostrados através de sensores de abordagem passiva, aplicados em variações contextuais relacionadas aos fatores de dependência. Sendo assim, inicialmente realizamos o levantamento das principais bases de dados utilizadas em estudos envolvendo aplicações de ITS. Baseando-se em revisões anteriores (GEYER et al., 2020), elencamos características necessárias para condução dos experimentos e avaliamos a aderência a elas dos conjuntos de dados disponíveis. Embora os conjuntos possuam outros sensores, nossa análise focou a abordagem passiva.

Na Tabela 8 é detalhada a comparação. São utilizados os marcadores *Sim (S)*, *Não (N)* e *Indefinido (I)*. São considerados *S* os conjuntos que cumprem todos os requisitos de determinada característica, *N* aqueles que não cumprem ao menos uma restrição, e *I* quando não foi possível identificar informações do parâmetro avaliado. Na avaliação dos sensores inerciais e magnetômetro, foi considerado como requisito os conjuntos possuírem dados brutos nos três eixos. Para diferentes veículos, o requisito foi de utilizar diferentes modelos. Para ambientes, considerou-se diferentes superfícies, com variação de pavimentações ou não pavimentação. Para colocações, o requisito torna necessário ao menos um conjunto de sensores inerciais abaixo da suspensão e um acima. Por fim, o posicionamento controlado é requerido por produzir dados de forma mais confiável e permitir análises de propriocepção e exterocepção.

Tabela 8 – Comparação de conjuntos de dados públicos

		Conjunto de Dados					
		<i>KITTI</i>	<i>Apollo Scape</i>	<i>nuScenes</i>	<i>Lyft Level 5</i>	<i>Waymo</i>	<i>A2D2</i>
<b>Sensores</b>	<i>Câmera</i>	S	S	S	S	S	S
	<i>GPS</i>	S	S	S	N	N	S
	<i>Acelerômetro 3D</i>	S	S	N	N	N	S
	<i>Giroscópio 3D</i>	S	S	S	N	N	N
	<i>Magnetômetro 3D</i>	N	N	N	N	N	N
<b>Contexto</b>	<i>Diferentes Veículos</i>	N	I	N	I	I	I
	<i>Diferentes Condutores</i>	I	I	I	I	I	I
	<i>Diferentes Ambientes</i>	I	I	I	I	I	I
	<i>Diferentes Colocações</i>	N	N	N	N	N	N
	<i>Posicionamento Controlado</i>	S	I	S	N	N	I

Fonte: Desenvolvido pelo autor.

Conforme observado na análise comparativa, nenhum dos conjuntos de dados disponíveis apresenta dados para todos os sensores necessários, dentre sensores inerciais e de suporte. Na análise de contexto as características avaliadas se mostram ainda menos presentes, especialmente por conta de os sensores inerciais não serem o foco de nenhum dos conjuntos e, portanto, sua abordagem de coleta não ser tratada de forma adequada. Em geral, as propriedades avaliadas como *I* são propensas a serem *N*, uma vez que sua não especificação tende a ter sido em

razão de não ser considerada ou não existir. Em uma análise ampla, nenhum dos conjuntos de dados avaliados atende os requisitos para experimentação deste estudo e, por isso, se fez necessário conduzir uma etapa de coleta de dados, detalhada nas próximas subseções.

## 5.1 REDE DE SENSORES

Para coletar os dados foram desenvolvidas duas redes de sensores, sendo cada uma delas composta por um SBC Raspberry Pi e três módulos MPU-9250, cada um deles equipado com um acelerômetro 3D, giroscópio 3D, magnetômetro 3D e um sensor de temperatura. Uma fonte externa de GPS também foi utilizada, produzindo dados de localização e velocidade, assim como uma câmera para captura de vídeo do ambiente. A Tabela 9 detalha o *hardware* utilizado. Embora o foco deste estudo esteja na utilização dos dados do acelerômetro, giroscópio e GPS, também foram coletados dados do magnetômetro, temperatura e vídeo para produzir conjuntos de dados completos de sensores passivos, úteis para pesquisas futuras.

Tabela 9 – Hardware da rede de sensores

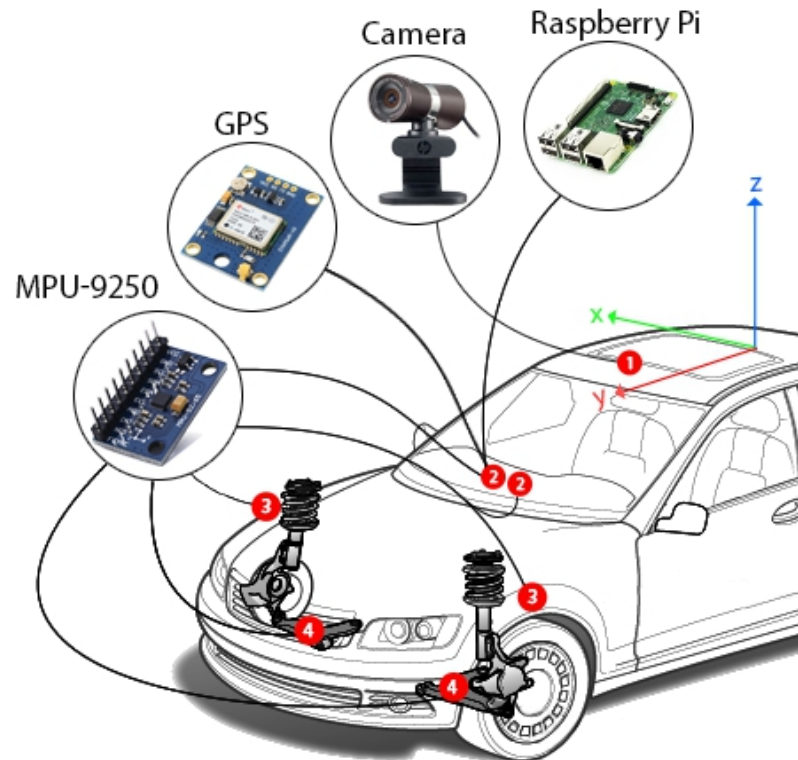
Hardware	Sensor	Dados	Taxa
HP Webcam HD-4110	Câmera	Vídeo 720p	30 Hz
Xiaomi Mi 8	GPS	Velocidade em $m/s$ , latitude, longitude, etc.	1 Hz
MPU-9250	Acelerômetro	Aceleração 3D em $m/s^2$	100 Hz
	Giroscópio	Taxa de rotação 3D em $graus/s$	
	Magnetômetro	Campo geomagnético ambiente 3D em $\mu T$	
	Temperatura	Temperatura em $^{\circ}C$	

Fonte: Desenvolvido pelo autor.

Todo o equipamento foi fixado no veículo conforme detalha a Figura 22. A câmera foi colocada na parte externa do teto do carro (1), capturando vídeo do ambiente em 30Hz. O receptor GPS foi colocado internamente no painel (2), amostrando dados em 1Hz. Os seis módulos MPU-9250 foram distribuídos no veículo de forma a considerar os dados provindos de pontos com diferentes influências da propriedade de dependência veicular. Sendo assim, em ambas as extremidades do eixo dianteiro (lado direito e esquerdo) foi anexado um módulo ao braço de controle (4), localizado abaixo e próximo à suspensão do veículo; outro módulo foi colocado acima e próximo da suspensão, anexado à carroceria imediatamente acima do pneu (3); e outro módulo foi anexado no painel do veículo (2), dentro da cabine. Em relação ao referencial de amostragem dos módulos MPU-9250, foi utilizada a abordagem de posicionamento controlado, onde a colocação dos módulos foi realizada de forma que os três eixos do sistema de coordenadas do sensor ficaram alinhados com os do veículo, sendo tanto referencial de coleta como de análise. O acelerômetro foi ajustado em um *Full Scale Range* (FSR) de 8g e o giroscópio de 1000 $graus/s$ , constituindo intervalos consistentes para não saturar, ambos amostrando em 100Hz (propriedades sensoriais). Estes valores foram obtidos através de experimentos exploratórios desta pesquisa.



Figura 22 – Colocação e posicionamento da rede de sensores no veículo



Fonte: Desenvolvido pelo autor.

## 5.2 EXECUÇÃO DA COLETA

Para o desenvolvimento e validação de modelos de percepção veicular adaptativa, se mostrou necessário produzir dados em variações contextuais para obter variabilidade de condições relacionadas aos fatores de dependência. Sendo assim, além da amostragem de velocidade (propriedade de condução) e da colocação dos sensores inerciais em diferentes pontos da estrutura veicular (propriedade veicular), nós realizamos diversas coletas de dados utilizando da rede de sensores em três diferentes modelos de veículos (propriedade veicular), com três diferentes motoristas variando a velocidade de 0 *km/h* até 91,98 *km/h* (propriedade de condução) trafegando em três cenários distintos (propriedade ambiental). Cada cenário contém vias sem pavimentação (terra) e trechos pavimentados (asfalto ou paralelepípedo), com variações gerais do ambiente, tais como presença de lombadas, buracos, diferentes níveis de conservação do pavimento, etc. Detalhes dos cenários são apresentados na próxima seção. Sendo assim, foram produzidos nove conjuntos de dados denominados *Passive Vehicular Sensors Dataset* (PVS 1-9), detalhados na Tabela 10. Cada um dos conjuntos de dados é composto pelos arquivos especificados na Tabela 11. Os dados foram coletados no município de Anita Garibaldi, no interior do estado de Santa Catarina, Brasil, entre os dias 24 e 26 de dezembro de 2019. O ponto médio dos cenários é dado pelas coordenadas (-27,69983094872972, -51,11577673365139).

Tabela 10 – Conjuntos de dados produzidos

Nome	Veículo	Condutor	Cenário
PVS 1	Volkswagen Saveiro	Condutor 1	Cenário 1
PVS 2	Volkswagen Saveiro	Condutor 1	Cenário 2
PVS 3	Volkswagen Saveiro	Condutor 1	Cenário 3
PVS 4	Fiat Bravo	Condutor 2	Cenário 1
PVS 5	Fiat Bravo	Condutor 2	Cenário 2
PVS 6	Fiat Bravo	Condutor 2	Cenário 3
PVS 7	Fiat Palio	Condutor 3	Cenário 1
PVS 8	Fiat Palio	Condutor 3	Cenário 2
PVS 9	Fiat Palio	Condutor 3	Cenário 3

Fonte: Desenvolvido pelo autor.

Tabela 11 – Arquivos que compõe cada conjunto de dados

Arquivo	Descrição
dataset_gps.csv	Dados de GPS, incluindo latitude, longitude, altitude, velocidade, acurácia, etc.
dataset_mpu_left.csv	Dados dos MPUs colocados no lado esquerdo do veículo, combinados com dados GPS.
dataset_mpu_right.csv	Dados dos MPUs colocados no lado direito do veículo, combinados com dados GPS.
dataset_labels.csv	Classes de dados para cada amostra do conjunto de dados (para ambos os lados).
dataset_mpu_left.csv	Dados dos MPUs colocados no lado esquerdo do veículo.
dataset_mpu_right.csv	Dados dos MPUs colocados no lado direito do veículo.
dataset_settings_left.csv	Configurações dos MPUs colocados no lado esquerdo do veículo. Inclui faixa de medição, resolução, etc.
dataset_settings_right.csv	Configurações dos MPUs colocados no lado direito do veículo. Inclui faixa de medição, resolução, etc.
map.html	Mapas interativos com os dados e as classes.
video_dataset_left.mp4	Vídeo com gráficos dos dados dos MPUs e GPS, amostrados no lado esquerdo do veículo.
video_dataset_right.mp4	Vídeo com gráficos dos dados dos MPUs e GPS, amostrados no lado direito do veículo.
video_environment.mp4	Vídeo do ambiente externo.
video_environment_dataset_left.mp4	Vídeos lado a lado de video_environment.mp4 e video_dataset_left.mp4
video_environment_dataset_right.mp4	Vídeos lado a lado de video_environment.mp4 e video_dataset_right.mp4

Fonte: Desenvolvido pelo autor.

### 5.3 CLASSES DE DADOS

Para criação dos rótulos correspondentes às classes de dados, foram utilizados GT de anotação humana e de anotação automatizada por máquina. Os primeiros rótulos criados foram os das classes de dados de tipo de superfície de pista. Para essas classes, foram criados os rótulos *terra* (*dirt\_road*), *paralelepípedo* (*cobblestone\_road*) e *asfalto* (*asphalt\_road*). A anotação foi realizada via GT humano, uma vez que esta classe representa uma característica objetiva. Desta forma, através da observação do que a roda toca por meio do vídeo capturado, os delimitadores de início e fim dessas classes puderam ser facilmente apontados. A Figura 23 ilustra os tipos de superfície presentes em todos os conjuntos de dados PVS, com a Figura 24 detalhando a distribuição delas no mapa de cada cenário. A Tabela 12 detalha as métricas para cada conjunto, com a quantificação das amostras e a proporção da distribuição das classes de dados, informações importantes para se analisar o nível de desbalanceamento das classes antes do treinamento dos modelos.

Figura 23 – Tipos de superfície presentes nos cenários



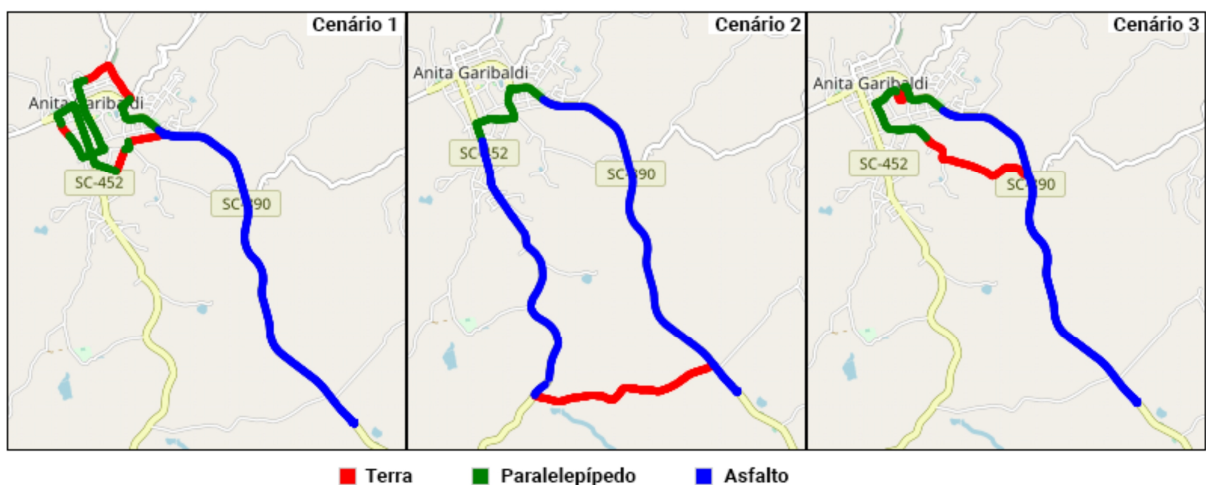
Terra (não pavimentado)

Paralelepípedo (pavimentado)

Asfalto (pavimentado)

Fonte: Desenvolvido pelo autor.

Figura 24 – Mapa das superfícies presentes nos cenários.



Fonte: Desenvolvido pelo autor.

Tabela 12 – Quantificação para as classes de dados de tipo de superfície

Cenário	Conjunto	Número de Amostras			Distribuição das Classes de Dados (%)			
		Terra	Paralelepípedo	Asfalto	Total	Terra	Paralelepípedo	Asfalto
1	PVS 1	25868	61659	56509	144036	17,96	42,81	39,23
	PVS 4	23903	57670	50919	132492	18,04	43,53	38,43
	PVS 7	23778	54224	50546	128548	18,50	42,18	39,32
2	PVS 2	44618	20737	59330	124685	35,78	16,63	47,58
	PVS 5	60539	18143	55195	133877	45,22	13,55	41,23
	PVS 8	44939	18825	59854	123618	36,35	15,23	48,42
3	PVS 3	28659	26143	51014	105816	27,08	24,71	48,21
	PVS 6	23888	31641	40750	96279	24,81	32,86	42,32
	PVS 9	23153	25182	43220	91555	25,29	27,50	47,21

Fonte: Desenvolvido pelo autor.

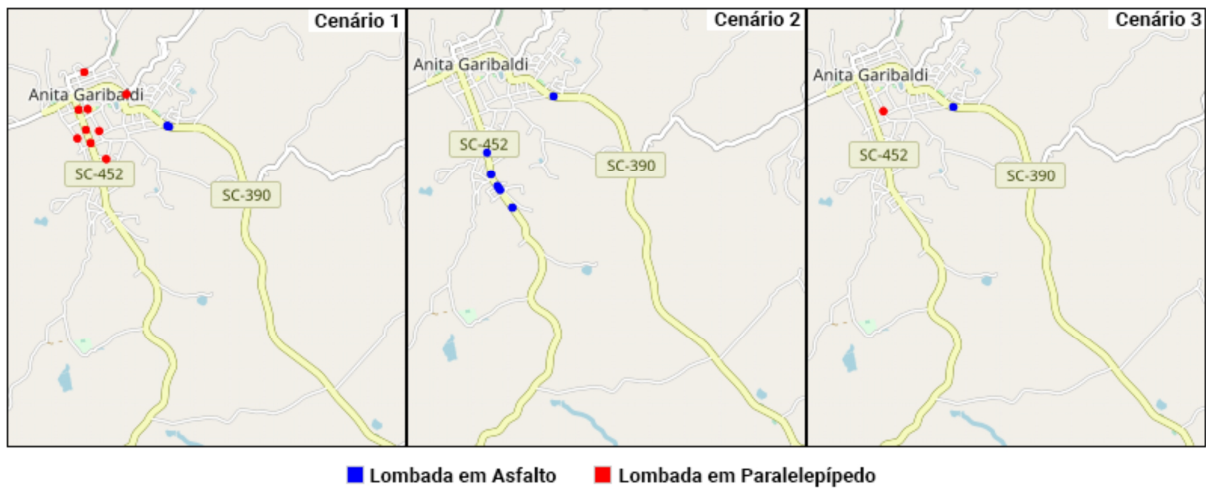
O segundo grupo de classes de dados rotuladas foi o de lombadas. Assim como as classes de tipo de superfície, as de lombadas correspondem a características objetivas e foram anotadas com GT humano. Foram criados os rótulos *com lombada* (*asphalt\_speed\_bump* e *cobblestone\_speed\_bump*), e *sem lombada* (*no\_speed\_bump*). As lombadas foram amostradas nos dois tipos de pavimentação, conforme ilustra a Figura 25. A Figura 26 detalha as localizações das lombadas, e a Tabela 13 especifica as métricas destas classes de dados.

Figura 25 – Lombadas em diferentes superfícies presentes nos cenários



Fonte: Desenvolvido pelo autor.

Figura 26 – Lombadas em diferentes superfícies presentes nos cenários



Fonte: Desenvolvido pelo autor.

Tabela 13 – Quantificação para as classes de dados de lombadas

Cenário	Conjunto	Número de Amostras			Distribuição das Classes de Dados (%)	
		Com Lombada	Sem Lombada	Total	Com Lombada	Sem Lombada
1	PVS 1	3455	140581	144036	2,39	97,60
	PVS 4	3134	129358	132492	2,37	97,63
	PVS 7	2876	125672	128548	2,24	97,76
2	PVS 2	2006	122679	124685	1,61	98,39
	PVS 5	1943	131934	133877	1,45	98,55
	PVS 8	1837	121781	123618	1,49	98,51
3	PVS 3	609	105207	105816	0,57	99,42
	PVS 6	606	95673	96279	0,63	99,37
	PVS 9	643	90914	91557	0,71	99,30

Fonte: Desenvolvido pelo autor.

Por fim, foram criados os rótulos para o nível de qualidade da superfície de pista. Para estas classes de dados, foram criados os rótulos *boa* (*good\_road*), *regular* (*regular\_road*) e *ruim* (*bad\_road*). Ao contrário das classes anteriores, as classes de qualidade podem tornar-se subjetivas de acordo com o GT utilizado. Inúmeros estudos rotulam a qualidade com base na percepção de qualidade de um usuário ou através da quantificação de irregularidades e obstáculos presentes no segmento de pista. Contudo, utilizar GT de anotação humana através das metodologias descritas acima torna as classes subjetivas, com viés a percepção de quem rotulou. Logo, para não haver viés através de GT humano, mostra-se necessário uma média amostral de uma quantidade estatisticamente relevante de opiniões de qualidade, sobre cada pequeno segmento de via. Isto demanda uma grande quantidade de *feedbacks*, tornando inviável devido ao alto custo. Sendo assim, de forma a tratar este problema, desenvolvemos um GT de anotação automatizada por máquina, baseado na análise de vibração.

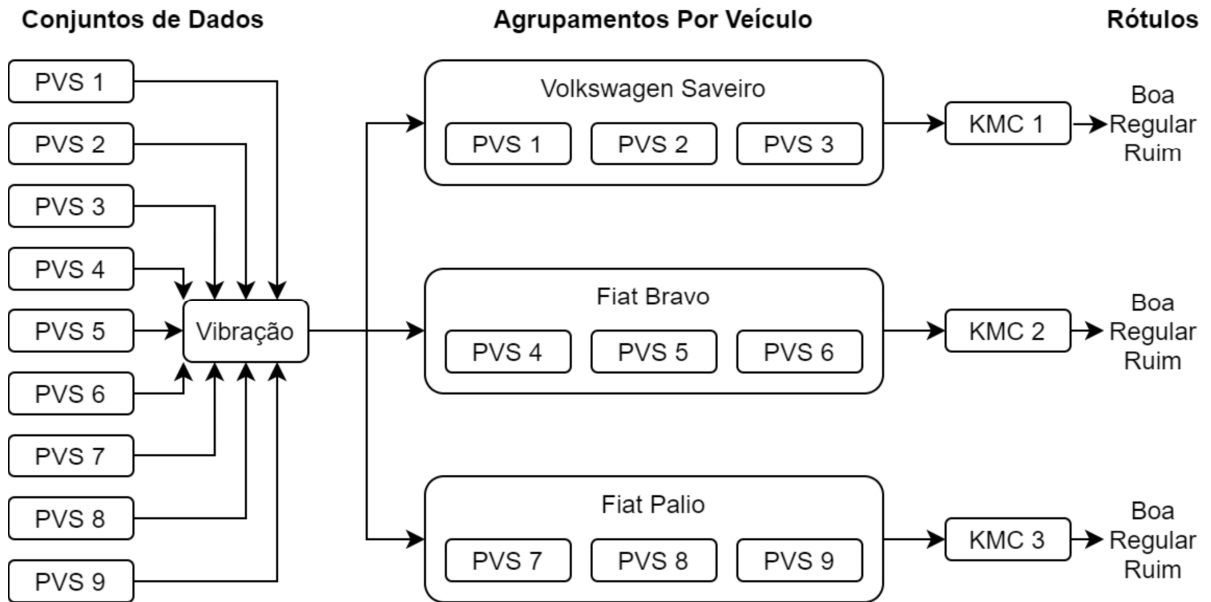
O processo de anotação de qualidade de superfície é ilustrado na Figura 27 e se inicia com a quantificação da irregularidade longitudinal dos segmentos analisados, baseado no cálculo de IRI de Li e Goldberg (2018), o qual utiliza dos dados do acelerômetro. A irregularidade calculada corresponde ao conjunto de desvios da superfície em relação a um plano de referência. Sendo assim, partindo de um *baseline* onde a variação dos sinais dos sensores inerciais é 0, caracterizando um segmento regular, quanto maior essa variação (vibração), mais irregular é a superfície. Para calcular a irregularidade, foi empregado RMS, a qual consiste de uma técnica estatística que mede a magnitude de uma variação, sendo especialmente útil quando os valores empregados alternam entre positivos e negativos, como é o caso dos sinais analisados.

A utilização do RMS com os dados dos sensores inerciais fornece a magnitude da irregularidade da superfície com a influência dos fatores de dependência. Sendo assim, é necessário considerar a forma com que cada um dos fatores influencia o valor resultante. Desta forma, após seu cálculo, a magnitude da irregularidade (propriedade ambiental) é normalizada pela velocidade acumulada no segmento de dados (propriedade de condução) (LI; GOLDBERG, 2018). Em suma, nesta primeira etapa, através de uma janela deslizante de 500 amostras, com 250 amostras à direita e outras 250 à esquerda, os sinais são aplicados ao RMS, o resultado multiplicado pela quantidade de amostras e normalizado pela velocidade acumulada no segmento. A equação abaixo detalha este cálculo, onde  $i$  é o índice da primeira amostra da janela,  $j$  o índice da última amostra,  $R$  o RMS dos sinais dos sensores inerciais, e  $v$  a velocidade.

$$Vibração = \frac{(j - i + 1) \cdot R}{\sum_{x=i}^j v_x} \cdot 100 \quad (5.1)$$

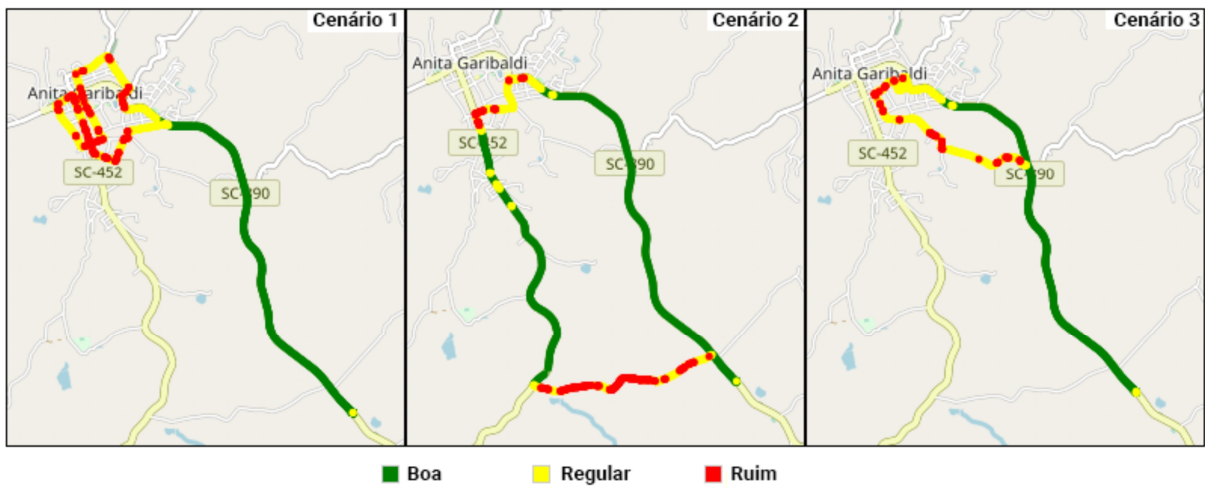
As características de vibração foram extraídas para cada um dos conjuntos PVS. Em seguida, estas características foram agrupadas de acordo com o modelo de veículo utilizado para considerar os fatores de dependência (propriedade veicular), onde no primeiro grupo ficaram as características dos PVS 1-3 (Volkswagen Saveiro), no segundo as dos PVS 4-6 (Fiat Bravo), e no terceiro as dos PVS 7-9 (Fiat Palio). Uma vez que os três cenários de coleta de dados contam com ambientes nos quais há segmentos nos três níveis de qualidade, e que cada um dos agrupamentos por veículo conta com dados dos três cenários, podemos concluir que necessariamente cada agrupamento possui características de alto nível que representam segmentos com pista de qualidade boa, regular e ruim. Sendo assim, em seguida cada agrupamento por veículo foi empregado separadamente em um modelo KMC de 3 *clusters*, os quais representam os níveis de qualidade, onde as características de vibração foram agrupadas de acordo com sua similaridade. O critério de parada foi o máximo de iterações, definido em 100.000. Os demais hiperparâmetros foram utilizados com valores padrões, encontrados em Scikit-Learn (2021a). Os agrupamentos obtidos através do modelo de KMC resultaram nos rótulos das classes de níveis de qualidade ilustrados na Figura 28. Uma vez que a qualidade da superfície depende diretamente do pneu que a toca, foram produzidos separadamente rótulos para os sensores colocados no lado direito do veículo, assim como aqueles do lado esquerdo. A Tabela 14 detalha as métricas dessas classes.

Figura 27 – Processo de GT de anotação automatizada por máquina para qualidade de superfície



Fonte: Desenvolvido pelo autor.

Figura 28 – Diferentes níveis de qualidade de superfície presentes nos cenários



Fonte: Desenvolvido pelo autor.

Tabela 14 – Quantificação para as classes de dados de qualidade de superfície

Cenário	Conjunto	Número de Amostras				Distribuição das Classes de Dados (%)		
		Boa	Regular	Ruim	Total	Boa	Regular	Ruim
1	PVS 1 Esquerda	56577	65304	22155	144036	39,28	45,34	15,38
	PVS 1 Direita	56595	66038	21403	144036	39,29	45,85	14,86
	PVS 4 Esquerda	50744	62838	18910	132492	38,30	47,43	14,27
	PVS 4 Direita	50732	64102	17658	132492	38,29	48,38	13,33
	PVS 7 Esquerda	49855	64743	13950	128548	38,78	50,36	10,85
	PVS 7 Direita	50040	67230	11278	128548	38,93	52,30	8,77
2	PVS 2 Esquerda	56086	37921	30677	124684	44,98	30,41	24,60
	PVS 2 Direita	55926	38057	30701	124684	44,85	30,52	24,62
	PVS 5 Esquerda	53715	32999	47163	133877	40,12	24,65	35,23
	PVS 5 Direita	53510	32587	47780	133877	39,97	24,34	35,69
	PVS 8 Esquerda	56466	29949	37203	123618	45,68	24,23	30,10
	PVS 8 Direita	56617	32301	34700	123618	45,80	26,13	28,07
3	PVS 3 Esquerda	50923	46638	8255	105816	48,12	44,07	7,80
	PVS 3 Direita	50906	47384	7526	105816	48,11	44,78	7,11
	PVS 6 Esquerda	40446	51051	4782	96279	42,01	53,02	4,97
	PVS 6 Direita	40453	50180	5646	96279	42,02	52,12	5,86
	PVS 9 Esquerda	42557	38919	10079	91555	46,48	42,51	11,01
	PVS 9 Direita	42711	42625	6219	91555	46,65	46,56	6,79

Fonte: Desenvolvido pelo autor.



## 6 CLASSIFICAÇÃO DE TIPO DE SUPERFÍCIE DE PISTA 1

Nesta seção é apresentado o primeiro de dois estudos voltados ao desenvolvimento de um modelo adaptativo para classificação do tipo da superfície de pista. Este estudo teve como objetivo experimentar e comparar modelos baseados em algumas das técnicas mais utilizadas nos estudos relacionados, com modelos baseados em *Deep Learning*. O processo de desenvolvimento e experimentação é detalhado nas próximas subseções. Na primeira delas, de pré-processamento, foi realizada a seleção de variáveis, normalização dos sinais, extração de características e separação de dados para treinamento e validação. O *design* experimental foi produzido de forma a construir experimentos que permitiram avaliar a capacidade de generalização do aprendizado de cada modelo para contextos desconhecidos e, portanto, avaliar sua adaptabilidade. Na segunda subseção, de processamento, foram desenvolvidos e testados 34 diferentes modelos para classificar a superfície de pista, dentre modelos de Aprendizado de Máquina clássico e *Deep Learning*. Por fim, na última subseção são detalhados e comparados os resultados obtidos.

### 6.1 PRÉ-PROCESSAMENTO

Com os conjuntos de dados criados, os dados brutos foram pré-processados antes de serem entregues aos modelos de Aprendizado de Máquina. Este estudo teve como foco a utilização dos valores amostrados no braço de controle, localizado abaixo e próximo ao sistema de suspensão. De acordo com o modelo QC, esses valores possuem interferência apenas da massa não suspensa por meio da rigidez e absorção do pneu (AL-YAFEAI; DARABSEH; MOURAD, 2019). Sendo assim, cada amostra utilizada possui os valores de força de aceleração e de taxa de rotação, ambos em três eixos, e o valor da velocidade. Neste estudo, utilizamos de ambos os sensores inercias uma vez que os consideramos complementares, com cada um deles fornecendo um tipo específico de dado acerca dos movimentos veiculares. Também consideramos todos os eixos dos sensores e não apenas o de maior interesse, como geralmente é feito nos estudos de exterocepção, uma vez que todos os eixos apresentam informações relevantes, que devem ser consideradas no desenvolvimento de um modelo mais confiável.

Após a seleção das sete variáveis, seus dados foram transformados para se adequar às entradas das técnicas de classificação de padrões. Para os modelos de Aprendizado de Máquina clássicos, foi necessário extrair em pré-processamento as características de alto nível que bem representem as classes de dados. Para isto, foram aplicadas estatísticas janeladas utilizando os métodos Desvio Padrão, Média e Variância para cada eixo dos sensores inerciais e Média para velocidade, os quais constituem os métodos mais comumente usados em estudos relacionados para extrair características baseadas em vibração de sinais de sensores inerciais (ALQUDAH; SABABHA, 2016; ANDRIA et al., 2016; BELLO-SALAU et al., 2018; BOSE et al., 2018; HOU et al., 2017; LI et al., 2016; LIMA et al., 2016; PHOLPRASIT; CHOOCHAIWATTANA; SAIPRASERT, 2015; PRAPULLA; RAO; HERUR, 2017; SAVERA et al., 2016; SINGH et

al., 2017). Para os modelos baseados em *Deep Learning*, uma vez que estas técnicas produzem melhor desempenho quando as variáveis são escaladas em uma faixa de valores, os dados foram normalizados com *Robust Scaler*, *Min Max Scaler* no intervalo [0,1] e *Min Max Scaler* no intervalo [-1,1], para avaliar o mais adequado. Para analisar a influência do número de amostras em todos os modelos de Aprendizado de Máquina clássico e de *Deep Learning*, foram criados os *experimentos por tamanho de janela de dados*, onde foram utilizadas janelas fixas sem sobreposição de 100, 150, 200, 250 e 300 amostras.

Após a transformação dos dados, definiu-se quais foram utilizados para treinamento e para validação. Embora um mesmo conjunto de dados seja comumente dividido em uma parte para treinamento e outra para validação (geralmente 70% - 30%), ou os conjuntos de dados sejam aplicados em uma metodologia de validação cruzada *k-fold*, ambas as abordagens no contexto deste estudo incorrem em viés por duas razões. A primeira, se todos os conjuntos de dados possuem uma parte para treinamento e outra para validação ou sejam distribuídos na abordagem *k-fold* de  $k$  igual a 9, isso implica que a técnica de reconhecimento de padrões terá para treinamento dados amostrados em todos os veículos, com todos os motoristas, e em todos os cenários. Isso, por si só, já leva a técnica a obter excelentes resultados, mas não nos diz nada sobre a capacidade de generalizar seu aprendizado de classificar padrões em contextos desconhecidos. A segunda razão, a qual incide na divisão de um mesmo conjunto PVS, decorre de que esse tipo de divisão pode levar a situações em que os dados de validação são compostos principalmente por dados com padrões mais facilmente reconhecidos, como segmentos de pavimento asfáltico, uma vez que a vibração do sinal neste tipo de superfície é muito menor do que nas outras. Portanto, para avaliar corretamente a generalização de cada técnica, avaliando sua confiabilidade em contextos desconhecidos (diferentes veículos, motoristas ou cenários/ambientes), os dados de treinamento e validação foram divididos de acordo com o conjunto de dados, separando-os em três *experimentos por contexto*:

**Experimento por Contexto 1:** O modelo aprende dados de todos os veículos e motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (65%):** PVS 1, PVS 3, PVS 4, PVS 6, PVS 7, PVS 9.
- **Validação (35%):** PVS 2, PVS 5, PVS 8.

**Experimento por Contexto 2:** O modelo aprende dados de todos os cenários para alguns veículos e alguns motoristas; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 3, PVS 7, PVS 8, PVS 9.
- **Validação (34%):** PVS 4, PVS 5, PVS 6.

**Experimento por Contexto 3:** O modelo aprende dados de alguns veículos com alguns motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 4, PVS 6, PVS 8, PVS 9.
- **Validação (34%):** PVS 3, PVS 5, PVS 7.

É importante observar que cada conjunto de dados consiste em um conjunto disjuncto de variáveis. Portanto, cada um deles contém dados para todas as variáveis de entrada do modelo (força de aceleração, taxa de rotação e velocidade) e a representação de todas as classes de dados (terra, paralelepípedo e asfalto). O que diferencia um conjunto de outro é o contexto de coleta de dados, onde há variação nas propriedades de dependência. A Figura 29 ilustra a divisão dos grupos de treinamento e validação de acordo com essas propriedades.

Figura 29 – Divisão dos conjuntos de dados PVS em grupos de treinamento e validação

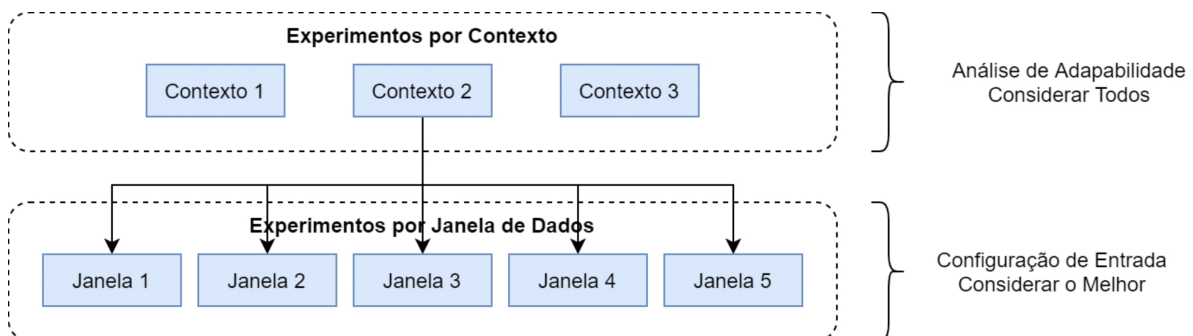
Experimento por Contexto	Veículo			Condutor			Cenário		
	Saveiro	Bravo	Palio	1	2	3	1	2	3
1	Blue	Blue	Blue	Blue	Blue	Blue	Green	Yellow	Green
2	Green	Yellow	Green	Green	Yellow	Green	Blue	Blue	Blue
3	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue

Treinamento e Validação
  Apenas em Treinamento
  Apenas em Validação

Fonte: Desenvolvido pelo autor.

Embora a abordagem de divisão de dados adotada não seja comum, ela foi necessária para permitir avaliar o comportamento dos modelos de Aprendizado de Máquina quando submetido a dados coletados em um veículo, motorista ou cenário desconhecido. Nos três experimentos, todas as classes de dados estão presentes na fase de treinamento e validação. Em resumo, cada experimento executado neste estudo consiste de um elemento do produto cartesiano entre *experimentos por contexto* e *experimentos por tamanho de janela de dados*, conforme ilustra a Figura 30.

Figura 30 – Combinação de tipos de experimentos



Fonte: Desenvolvido pelo autor.

Para avaliar a necessidade de aplicar técnicas de balanceamento de classe de dados, foi medido a distribuição de cada classe conforme detalhado na Tabela 15. A distribuição de classes foi calculada com os dados de treinamento (HE; MA, 2013; KUHN; JOHNSON et al., 2013), uma vez que estes são os dados utilizados pelos modelos para aprender os padrões. A distribuição foi calculada em relação ao número de amostras, uma vez que as janelas de dados são definidas de acordo com esse parâmetro.

Tabela 15 – Distribuição de classes de dados de tipo de superfície de pista

Fonte de Dados	Classe de Dados					
	Terra		Paralelepípedo		Asfalto	
	Percentual	Proporção	Percentual	Proporção	Percentual	Proporção
Exp. por Contexto 1	21,36%	1:3,7	36,71%	1:1,7	41,92%	1:1,4
Exp. por Contexto 2	26,59%	1:2,8	28,78%	1:2,5	44,61%	1:1,2
Exp. por Contexto 3	26,15%	1:2,8	30,26%	1:2,3	43,58%	1:1,3

Fonte: Desenvolvido pelo autor.

Analisando a Tabela 15, observamos que a distribuição das classes de tipo de superfície de pista situa-se entre 21,36% e 44,61%, dependendo do experimento. A proporção varia entre 1:1,2 a 1:3,7, onde para 1 amostra de uma determinada classe de dados, existem de 1,2 a 3,7 amostras nas outras classes. De acordo com (FERNÁNDEZ et al., 2018), um conjunto de dados é considerado desbalanceado quando existe uma desproporção significativa, ou em alguns casos severa, entre o número de amostras de cada classe. O desbalanceamento de classes pode ser considerado leve ou severo, onde as proporções de distribuição que variam de 1:4 até 1:100 (presença de 20% - 1%) são consideradas desbalanceamento leve e proporções de distribuição que variam de 1:100 ou mais (<1% de presença) são consideradas desbalanceamentos severos (KRAWCZYK, 2016; BROWNLEE, 2020). Como podemos observar, as proporções de distribuição das classes de dados neste estudo não é classificada sequer como desbalanceamento leve, pois em seu pior caso, a proporção 1:3.7 ainda é uma distribuição mais uniforme do que 1:4. De acordo com (BROWNLEE, 2020), o desbalanceamento leve geralmente não é uma preocupação, e o problema pode ser tratado como um problema de modelagem preditiva ou classificação normal. Sendo assim, se desbalanceamentos leves não dependem de balanceamento de classes, distribuições mais uniformes, como é o caso deste estudo, não necessitam de aplicação desta técnica.

## 6.2 PROCESSAMENTO

Após a etapa de pré-processamento, os dados foram utilizados em técnicas de Inteligência Artificial para classificação do tipo de superfície de pista. Por meio de nossa RSL sobre percepção veicular, identificamos as principais técnicas utilizadas na exerocepção veicular com sensores inerciais, a qual tem como objetivo reconhecer as características do ambiente externo

como tipo de superfície, qualidade do pavimento, buracos, lombadas, etc. Todos os modelos de Aprendizado de Máquina empregados em estudos anteriores na área são baseados em técnicas clássicas. Portanto, neste estudo foi realizada uma comparação entre as técnicas mais utilizadas na área, todas elas sendo o Aprendizado de Máquina clássico, com as técnicas de Aprendizado Profundo (*Deep Learning*), as quais ainda não foram utilizadas neste tipo de problema de classificação.

Dentre as técnicas de Aprendizado de Máquina clássico mais aplicadas na área, estão as três experimentadas neste estudo: KMC, SVM e KNN. Para a técnica de KMC, foram desenvolvidos modelos com 3 *clusters*, representando as classes a serem agrupadas. Para a técnica de SVM, foi adotado o esquema *one-vs-one* para classificação multiclasse, e experimentados modelos com três diferentes *kernels*: polinomial de grau 3, rbf e sigmoid. Por fim, para a técnica de KNN, de forma a identificar o valor ótimo de vizinhos, foram desenvolvidos modelos com 1, 2, 5, 10, 50, 100, 250, 500, 1000 vizinhos. Os demais hiperparâmetros das técnicas clássicas foram utilizados com seus valores padrões, disponíveis em Scikit-Learn (2021a), Scikit-Learn (2021b) e Scikit-Learn (2021c). No desenvolvimento dos modelos de *Deep Learning*, uma vez que nenhum estudo de exterocepção aplica este tipo de técnica, os modelos produzidos foram baseados em modelos de outros domínios que também utilizam de sensores inerciais para reconhecer padrões. Sendo assim, as Redes Neurais Profundas (*Deep Neural Networks - DNN*) desenvolvidas foram baseadas em estudos de reconhecimento de atividade humana (DEEP; ZHENG, 2019; ALEMAYOH; LEE; OKAMOTO, 2019; CHEN; XUE, 2015; YANG et al., 2018; ZEBIN et al., 2018; ZEBIN et al., 2019; WANG et al., 2019; AHMAD; KAZMI; ALI, 2019), estimativa de velocidade de caminhada (SHRESTHA; WON, 2018) e classificação de terrenos durante corrida humana (DIXON et al., 2019). Foram construídos e analisados modelos baseados em LSTM, CNN e CNN-LSTM. Todos os modelos de DNN são sequenciais e utilizam o otimizador Adam em conjunto com a função de perda de Entropia Cruzada Categórica, uma vez que se trata de um problema de classificação multiclasse.

Para as redes baseadas em LSTM, vários modelos foram desenvolvidos, dentre *vanilla* LSTM e *stacked* LSTM, tanto na forma unidirecional quanto bidirecional, detalhados na Tabela 16. O LSTM 7 foi o modelo desenvolvido com melhores resultados, consistindo de uma LSTM *stacked* unidirecional detalhada na Figura 31. Neste modelo, a DNN recebe um tensor de entrada *janelas x sequências x características*, onde *janelas* são os agrupamentos de janelas, *sequências* são os dados que uma janela possui, ou seja, a sequência de amostras, e *características* são os valores das 7 variáveis dos sensores, sendo assim, os valores de cada amostra. O modelo é composto por uma camada de entrada, três blocos de recorrência e regularização, e um bloco de camadas totalmente conectadas para produção de saída. As camadas de LSTM com 100 unidades são utilizadas para aprender as dependências temporais de longo prazo na sequência de dados. A regularização é feita por duas camadas, sendo *Batch Normalization* para padronizar as entradas de uma nova camada, reduzindo o tempo de treinamento e melhorando o desempenho (ZEBIN et al., 2018); e *Dropout* em 50% para evitar *overfitting*, ignorando neurônios selecionados aleatoriamente durante o treinamento. Após o processamento nas ca-

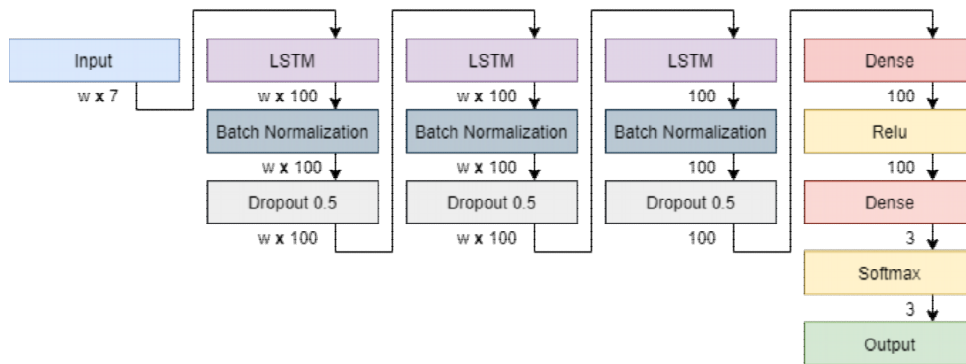
madras recorrentes, os parâmetros são passados para duas camadas *Dense*, a primeira com 100 neurônios com ativação *Relu*, e a segunda com 3 neurônios e ativação do *Softmax*, produzindo a classificação. A saída esperada da rede são os rótulos correspondentes à última amostra da janela.

Tabela 16 – Modelos de LSTM para classificação do tipo de superfície de pista

Nome	Camadas
LSTM 1	1 LSTM com 100 unidades, 1 Dropout de 0,5, 1 Dense com 100 unidades e ativação Relu, 1 Dense com 3 unidades e ativação Softmax.
LSTM 2	Igual a LSTM 1, mas conectando as sequencias <i>flattened</i> retornadas pela LSTM diretamente na camada Dense.
LSTM 3	Igual a LSTM 1, mas com LSTM bidirecional.
LSTM 4	Igual a LSTM 2, mas com LSTM bidirecional.
LSTM 5	3 blocos de LSTM com 100 unidades e Dropout em 0,5, 1 Dense com 100 unidades e ativação Relu, 1 Dense com 3 unidades e ativação Softmax.
LSTM 6	Igual a LSTM 5, mas com LSTM bidirecional e Dropout em 0,2.
LSTM 7	3 blocos de LSTM com 100 unidades, Batch Normalization e Dropout 0,5, 1 Dense com 100 unidades e ativação Relu, 1 Dense com 3 unidades e ativação Softmax.

Fonte: Desenvolvido pelo autor.

Figura 31 – Melhor modelo de LSTM para classificação de superfície de pista



Fonte: Desenvolvido pelo autor.

Para as redes baseadas em CNN, foram produzidos modelos de CNN *Stacked* utilizando diferentes camadas de *pooling*, conforme detalha a Tabela 17. O modelo desenvolvido com melhores resultados foi o CNN 8, detalhado na Figura 32. Esta DNN recebe um tensor de entrada *janelas x sequências x características* semelhante ao das redes baseadas em LSTM, o qual é processado por três blocos de camadas de convolução e regularização, e um bloco de camadas totalmente conectadas. Os blocos de extração de características utilizam nos sinais *kernels* de convolução de tamanho 3, com a primeira camada possuindo 64 filtros e as demais com 32 filtros. A regularização é feita pelas camadas *Batch Normalization* e *Dropout* em 15% e 20%. O último bloco de extração de características também possui uma camada *Global Average Pooling 1D* para extrair características mais robustas por valores médios em cada região, acelerando o processo de treinamento e evitando *overfitting* (YANG et al., 2018; WANG et al.,

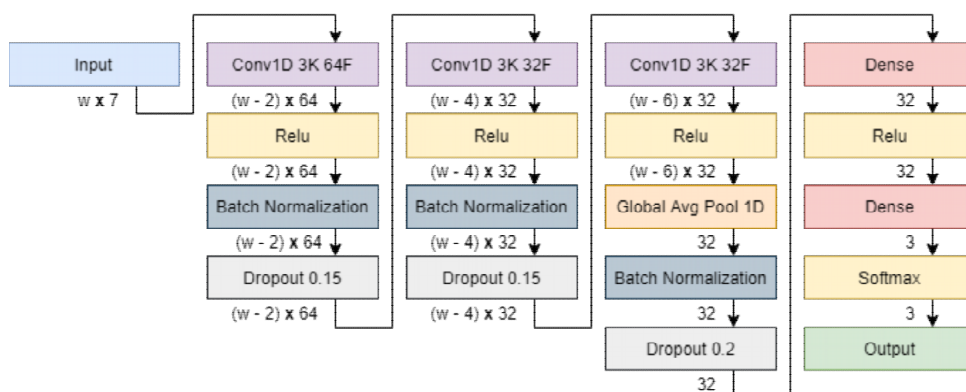
2019). Por fim, o último bloco consiste de duas camadas *Dense*, uma com 32 neurônios e ativação de *Relu*, e a outra com 3 neurônios e ativação de *Softmax*. A saída esperada da rede são os rótulos mais presentes na janela analisada.

Tabela 17 – Modelos de CNN para classificação do tipo de superfície de pista

Nome	Camadas
CNN 1	3 blocos de Conv1D com 64-64-128 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Flatten, 1 Dense com 100 unidades e ativação Relu, 1 Dense com 3 unidades e ativação Softmax.
CNN 2	Igual a CNN 1, mas com 1 Max Pooling 1D com <i>pool</i> de tamanho 2 depois dos blocos de convolução.
CNN 3	1 Conv1D com 64 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Batch Normalization, 1 Dropout em 0,15, 1 Conv1D com 32 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Global Avg Pool 1D, 1 Batch Normalization, 1 Dropout 0,2 e 1 Dense com 3 unidades e ativação Softmax.
CNN 4	2 Conv1D com 100 filtros, <i>kernel</i> de tamanho 10 e ativação Relu, 1 Max Pooling 1D com <i>pool</i> de tamanho 3, 2 Conv1D com 160 filtros, <i>kernel</i> de tamanho 10 e ativação Relu, 1 Global Avg Pool 1D, 1 Dropout 0,5 e 1 Dense com 3 unidades e ativação Softmax.
CNN 5	1 Conv1D com 64 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Max Pooling 1D com <i>pool</i> de tamanho 2, 1 Conv1D com 64 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Flatten e 1 Dense com 3 unidades e ativação Softmax.
CNN 6	1 Conv1D com 24 filtros, <i>kernel</i> de tamanho 8 e ativação Relu, 1 Batch Normalization, 1 Spatial Dropout 0,15, 1 Conv1D com 12 filtros, <i>kernel</i> de tamanho 8 e ativação Relu, 1 Global Avg Pool 1D, 1 Batch Normalization, 1 Dropout 0,2, 1 Dense com 48 unidades e ativação Relu, 1 Batch Normalization, 1 Dropout em 0,25, 1 Dense com 3 unidades e ativação Softmax.
CNN 7	3 blocos de Conv1D com 128-64-32 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Batch Normalization e 1 Max Pooling 1D com <i>pool</i> de tamanho 2, 1 Flatten, 1 Dropout em 0,2, 1 Dense com 3 unidades e ativação Softmax.
CNN 8	2 blocos de Conv1D com 64-32 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Batch Normalization e 1 Dropout em 0,15, 1 Conv1D 32 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Global Avg Pool 1D, 1 Batch Normalization, 1 Dropout em 0,2, 1 Dense com 32 unidades e ativação Relu e 1 Dense com 3 unidades e ativação Softmax.

Fonte: Desenvolvido pelo autor.

Figura 32 – Melhor modelo de CNN para classificação do tipo de superfície de pista



Fonte: Desenvolvido pelo autor.

Para as redes baseadas em CNN-LSTM, foram produzidos modelos híbridos detalhados na Tabela 18. O modelo desenvolvido com melhores resultados foi o CNN-LSTM 6, detalhado na Figura 33. Esta DNN recebe um tensor *janelas x sequências x subsequências x*

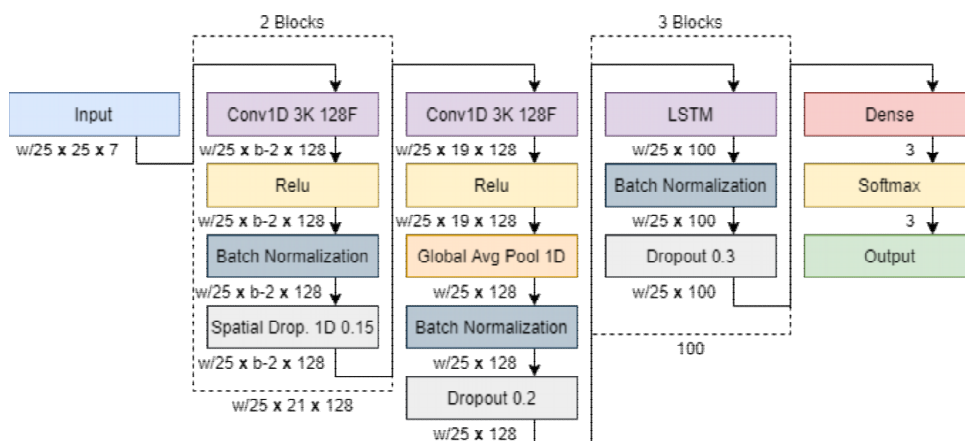
*características*, onde *janelas* são os agrupamentos de janelas, *sequências* são as janelas de dados, *subsequências* são os subgrupos de janelas e *características* são os valores das 7 variáveis dos sensores. Os dados são inicialmente processados por três blocos de camadas de convolução e regularização, três blocos de camadas recorrentes e de regularização, e um bloco de camadas totalmente conectadas. Os blocos de extração de características utilizam nos sinais *kernels* de convolução de tamanho 3 com 128 filtros. A regularização é feita pelas camadas de *Batch Normalization* e *Spatial Dropout 1D* em 15% e 20%. O último bloco possui uma camada *Global Average Pooling 1D*. Os blocos recorrentes possuem camadas de LSTM com 100 unidades e regularização por *Batch Normalization* e *Dropout* em 50%. Finalmente, os parâmetros resultantes são entregues a uma camada *Dense* com 3 neurônios e ativação do *Softmax*, produzindo a classificação. A saída esperada da rede são os rótulos correspondentes à última amostra na janela.

Tabela 18 – Modelos de CNN-LSTM para classificação do tipo de superfície de pista

Nome	Camadas
CNN-LSTM 1	2 Conv1D com 64 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, 1 Dropout em 0,5, 1 Max Pooling 1D com <i>pool</i> de tamanho 2, 1 Flatten, 1 LSTM com 100 unidades, 1 Dropout em 0,2, 1 Dense com 100 unidades e ativação Relu, 1 Dense com 3 unidades e ativação Softmax.
CNN-LSTM 2	Igual a CNN-LSTM 1, mas com 3 camadas Conv1D.
CNN-LSTM 3	Igual a CNN-LSTM 2, mas com 3 blocos de LSTM e Dropout após a camada Max Pooling 1D.
CNN-LSTM 4	Igual a CNN-LSTM 3, mas com os blocos LSTM sem camada Dropout.
CNN-LSTM 5	2 blocos de Conv1D com 128 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, Max Pooling 1D com <i>pool</i> de tamanho 2 e Batch Normalization, 1 bloco de 1 Conv1D com 128 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, Global Avg Pool 1D e Batch Normalization, 3 blocos de 1 LSTM com 100 unidades, Batch Normalization e Dropout em 0,3, 1 Dense com 3 unidades e ativação Softmax.
CNN-LSTM 6	2 blocos de 1 Conv1D com 128 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, Batch Normalization, e Spatial Dropout em 0,15, 1 bloco de Conv1D com 128 filtros, <i>kernel</i> de tamanho 3 e ativação Relu, Global Avg Pool 1D, Batch Normalization e Dropout em 0,2, 3 blocos de LSTM com 100 unidades, Batch Normalization e Dropout em 0,3, 1 Dense com 3 unidades e ativação Softmax.

Fonte: Desenvolvido pelo autor.

Figura 33 – Melhor modelo de LSTM-CNN para classificação do tipo de superfície de pista



Fonte: Desenvolvido pelo autor.



Os demais hiperparâmetros das técnicas de *Deep Learning* foram utilizados com seus valores padrões, disponíveis em TensorFlow (2021d) e TensorFlow (2021a).

### 6.3 ANÁLISE DE RESULTADOS

Todos os modelos desenvolvidos neste estudo foram programados em Python 3. Os modelos Aprendizado de Máquina clássico usaram a biblioteca Scikit-Learn, enquanto os modelos *Deep Learning* utilizaram o *framework* Keras com o *backend* Tensorflow. Os experimentos foram realizados no Google *Collaboratory* (Google Colab), com uma GPU NVIDIA Tesla P100 com 12 GB e 25 GB de RAM. Todas as configurações de treinamento estão detalhadas nos códigos-fonte documentados disponíveis na página do projeto no Github. Em todas as técnicas, exceto KNN que foi até a estabilidade, os experimentos foram executados 3 vezes a fim de minimizar a aleatoriedade dos parâmetros iniciais, recuperando apenas o melhor entre os três. Os resultados obtidos na aplicação do Aprendizado de Máquina clássico são detalhados nas Tabelas 19, 20 e 21. O valor detalhado corresponde à média de acurácia em validação dos três *experimentos por contexto* para um determinado *experimento por tamanho de janela de dados* e configuração dos hiperparâmetros da técnica.

Tabela 19 – Média de acurácia em validação para os modelos de KMC

Experimentos por Tamanho de Janela de Dados						
Clusters	100	150	200	250	300	Média
3	56,20%	58,13%	59,10%	59,67%	60,42%	58,70%

Fonte: Desenvolvido pelo autor.

Tabela 20 – Média de acurácia em validação para os modelos de SVM

Experimentos por Tamanho de Janela de Dados						
Kernel	100	150	200	250	300	Média
poly 3	65,00%	66,60%	68,05%	68,07%	68,71%	67,28%
rbf	71,95%	72,28%	72,68%	72,42%	72,45%	72,36%
sigmoid	52,80%	47,08%	57,50%	54,73%	56,75%	53,77%
<b>Média</b>	63,25%	61,99%	66,08%	65,08%	65,97%	64,47%

Fonte: Desenvolvido pelo autor.

Tabela 21 – Média de acurácia em validação para os modelos de KNN

Experimentos por Tamanho de Janela de Dados						
Vizinhos	100	150	200	250	300	Média
1	72,28%	72,77%	<b>74,46%</b>	73,45%	73,83%	73,36%
2	59,80%	61,08%	<b>62,69%</b>	60,87%	61,92%	61,27%
5	72,37%	72,78%	<b>74,79%</b>	73,80%	73,97%	73,54%
10	67,97%	68,39%	<b>70,23%</b>	68,81%	69,72%	69,02%
50	70,83%	71,54%	<b>72,01%</b>	71,30%	69,66%	71,07%
100	69,40%	<b>69,75%</b>	69,47%	68,77%	67,63%	69,00%
250	<b>66,25%</b>	65,18%	63,84%	62,61%	61,24%	63,83%
500	<b>62,15%</b>	59,57%	57,66%	54,65%	53,93%	57,59%
1000	<b>54,93%</b>	51,42%	49,12%	45,83%	44,05%	49,07%
<b>Média</b>	66,22%	65,83%	66,03%	64,45%	63,99%	65,31%

Fonte: Desenvolvido pelo autor.

Para a técnica KMC, o melhor modelo resultante para 3 *clusters* foi o de janela com 300 amostras, atingindo 58,24% de acurácia em treinamento e 60,42% em validação. Nessa técnica, quanto maior a janela de dados aplicada, maiores os valores de acurácia obtidos. Na técnica SVM, o *kernel* sigmoid obteve os piores valores de acurácia para todas as janelas de dados, seguido pelo *kernel* polinomial de grau 3. Sendo assim, o *kernel* rbf obteve os melhores valores de acurácia para todas as variações de tamanho de janela, onde o modelo com 200 amostras obteve acurácia de 76,15% para treinamento e 72,68% para validação. Neste *kernel*, o tamanho da janela de dados tem pouca influência no resultado final. Por fim, os modelos de KNN apresentaram melhores resultados com o número de vizinhos entre 1-50, onde o tamanho da janela também não tem muita influência no resultado. Com grande número de vizinhos, os resultados tendem a apresentar uma variação maior de acordo com o número de amostras na janela. Para o número de vizinhos entre 1-50, a janela com 200 amostras apresentou os melhores resultados, situando o melhor modelo com 5 vizinhos, onde obteve acurácia no conjunto de dados de treinamento de 85,90% e 74,79% na validação. Com esses resultados, os melhores modelos de cada técnica clássica são detalhados na Tabela 22, separados por *experimento por contexto*.

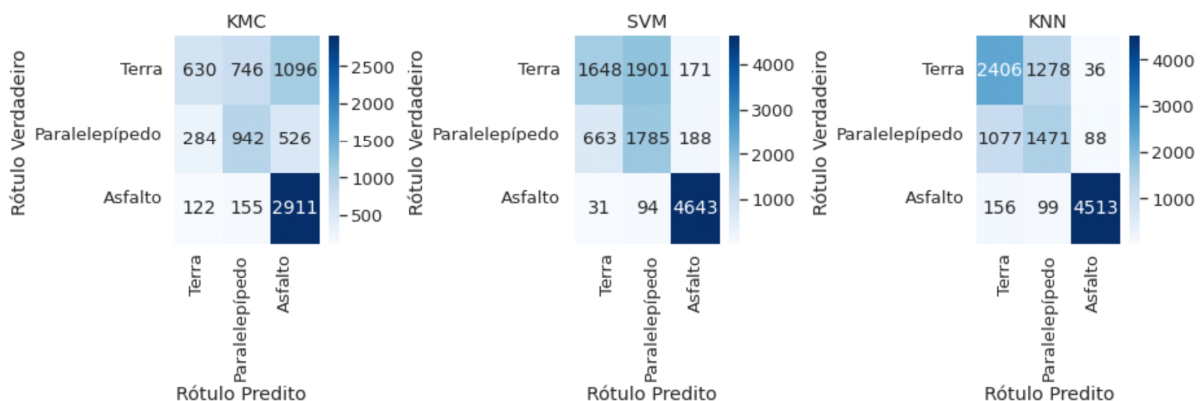
Tabela 22 – Valores de acurácia para os melhores modelos baseados em técnicas clássicas de aprendizado de máquina

Experimentos por Contexto					
Modelo	Fase	1	2	3	Média
KMC	Treinamento	60,53%	56,04%	58,13%	58,24%
	Validação	65,67%	60,32%	55,26%	60,42%
SVM	Treinamento	75,59%	77,03%	75,84%	76,15%
	Validação	67,28%	75,61%	75,16%	72,68%
KNN	Treinamento	84,50%	86,44%	86,76%	85,90%
	Validação	76,28%	71,11%	76,98%	<b>74,79%</b>

Fonte: Desenvolvido pelo autor.

Observando os resultados da Tabela 22, é possível perceber que a técnica de KNN não somente apresenta a maior média de acurácia, como também é a técnica mais estável, com menor variação de resultado entre experimentos com diferentes contextos. O KMC apresenta variância de 18,07%, o SVM de 14,63%, e o KNN de 6,85%. Além disso, observando a matriz confusão dos melhores modelos na Figura 34, a KNN é a única técnica em que o melhor modelo possui mais VP e VN que FP e FN, quando considerado cada tipo de superfície separadamente. Sendo assim, conforme detalhado na Tabela 23, KNN tem os melhores valores de *f1-score* para duas das três classes de dados, classificando segmentos de terra com *f1-score* de 65,39%, paralelepípedos com 53,65% e asfalto com 95,97%.

Figura 34 – Matriz de confusão para cada melhor modelo de técnicas clássicas de aprendizado de máquina



Fonte: Desenvolvido pelo autor.

Tabela 23 – Métricas de avaliação para cada melhor modelo baseado em técnicas clássicas de aprendizado de máquina

Modelo	Classe de Dados	F1-Score	Precisão	Recall
KMC	Asfalto	75,40%	64,22%	91,31%
	Paralelepípedo	52,41%	51,11%	53,77%
	Terra	35,92%	60,81%	25,49%
SVM	Asfalto	95,05%	92,82%	97,38%
	Paralelepípedo	55,64%	47,22%	67,71%
	Terra	54,37%	70,37%	44,30%
KNN	Asfalto	95,97%	97,33%	94,65%
	Paralelepípedo	53,65%	51,65%	55,80%
	Terra	65,39%	66,12%	64,68%

Fonte: Desenvolvido pelo autor.

Para os experimentos com as técnicas de *Deep Learning*, inicialmente foram avaliados os diferentes métodos para normalização dos sinais. Dentre os três experimentados, o *Min Max Scaler* entre  $[0,1]$  apresentou piores resultados, atingindo um maior valor de perda e uma acurácia menor. O *Robust Scaler* foi o segundo melhor, mantendo o sinal nos dados, mas sem escalar em um intervalo fixo. Por fim, os dados normalizados com *Min Max Scaler* entre  $[-1,1]$ , mantendo o sinal e escalando em um intervalo fixo, apresentaram convergência mais rápida, menor perda e maior acurácia. Sendo assim, os escaladores que mantêm o sinal obtiveram melhores resultados, uma vez que o sinal é importante neste estudo, indicando a informação de direção e não apenas uma diminuição de valor. Utilizando os dados normalizados, foram experimentados os modelos de redes propostos. Os resultados são detalhados nas Tabelas 24, 25 e 26.

Tabela 24 – Média de acurácia em validação para redes baseadas em LSTM

Modelo	Experimentos por Tamanho de Janela de Dados					Média
	100	150	200	250	300	
LSTM 1	89,14%	89,44%	<b>89,86%</b>	86,30%	87,59%	88,47%
LSTM 2	<b>87,38%</b>	87,33%	86,80%	86,26%	85,85%	86,72%
LSTM 3	89,11%	89,61%	89,99%	89,21%	<b>90,45%</b>	89,67%
LSTM 4	87,12%	<b>87,31%</b>	87,07%	86,11%	86,48%	86,82%
LSTM 5	89,74%	89,14%	90,15%	<b>90,60%</b>	89,63%	89,85%
LSTM 6	88,66%	89,69%	<b>90,41%</b>	88,99%	90,88%	89,72%
LSTM 7	90,39%	91,37%	91,85%	92,71%	<b>92,73%</b>	91,81%
<b>Média</b>	88,79%	89,13%	89,45%	88,60%	89,09%	89,01%

Fonte: Desenvolvido pelo autor.

Tabela 25 – Média de acurácia em validação para redes baseadas em CNN

Experimentos por Tamanho de Janela de Dados						
Modelo	100	150	200	250	300	Média
CNN 1	87,44%	87,60%	<b>88,25%</b>	87,07%	87,54%	87,58%
CNN 2	87,58%	87,82%	88,25%	<b>88,65%</b>	87,94%	88,05%
CNN 3	89,98%	91,67%	92,91%	92,81%	<b>93,02%</b>	92,08%
CNN 4	88,88%	90,20%	91,13%	91,78%	<b>92,00%</b>	90,80%
CNN 5	86,87%	87,71%	<b>88,61%</b>	87,81%	88,35%	87,87%
CNN 6	90,09%	91,12%	91,61%	<b>92,17%</b>	91,94%	91,39%
CNN 7	89,17%	89,59%	<b>90,14%</b>	89,66%	89,65%	89,64%
CNN 8	91,02%	91,83%	92,89%	92,99%	<b>93,17%</b>	92,38%
<b>Média</b>	88,88%	89,69%	90,47%	90,37%	90,45%	89,97%

Fonte: Desenvolvido pelo autor.

Tabela 26 – Média de acurácia em validação para redes baseadas em CNN-LSTM

Experimentos por Tamanho de Janela de Dados						
Modelo	100	150	200	250	300	Média
CNN-LSTM 1	87,56%	89,15%	89,35%	<b>90,97%</b>	90,78%	89,56%
CNN-LSTM 2	87,92%	88,38%	89,76%	89,54%	<b>90,47%</b>	89,21%
CNN-LSTM 3	87,45%	88,77%	90,09%	89,79%	<b>91,04%</b>	89,43%
CNN-LSTM 4	87,43%	89,43%	89,51%	89,93%	<b>90,99%</b>	89,46%
CNN-LSTM 5	90,03%	90,69%	91,23%	91,54%	<b>92,71%</b>	91,24%
CNN-LSTM 6	90,27%	91,50%	92,10%	92,48%	<b>92,77%</b>	91,82%
<b>Média</b>	88,44%	89,65%	90,34%	90,71%	91,46%	90,12%

Fonte: Desenvolvido pelo autor.

Todos os modelos de DNN propostos, mesmo nos piores resultados, apresentaram valores de acurácia em treinamento e validação maiores do que os obtidos pelo melhor modelo baseado em Aprendizado de Máquina clássico. Isso evidencia a capacidade do *Deep Learning* de aprender relações mais complexas entre os dados, em comparação às técnicas clássicas. Em todos os experimentos, o impacto na acurácia dado o tamanho da janela das amostras é muito pequeno, enquanto que as técnicas clássicas possuem grande variação. Em todas as abordagens de *Deep Learning* experimentadas, o uso da camada de *Bacth Normalization*, além de acelerar o treinamento, melhorou os resultados em todas as execuções. O uso da camada *Dropout* também se mostrou importante para a generalização do modelo. Nos modelos baseados em LSTM, a utilização de camadas bidirecionais praticamente não alterou os resultados, tendo aumento de acurácia para algumas janelas, e diminuição para outras, sempre em valores praticamente irrelevantes e, na média, menores que 1%. O uso de vetores de sequências retornados pela LSTM conectados diretamente nas camadas totalmente conectadas *Dense* piorou os resultados em todos os experimentos, diminuindo o valor de acurácia em até 4%. Sendo assim, o modelo

baseado em LSTM com os melhores resultados é o LSTM 7, em uma janela de 300 amostras, onde atingiu um valor de acurácia em treinamento de 96,67% e 92,73% em validação.

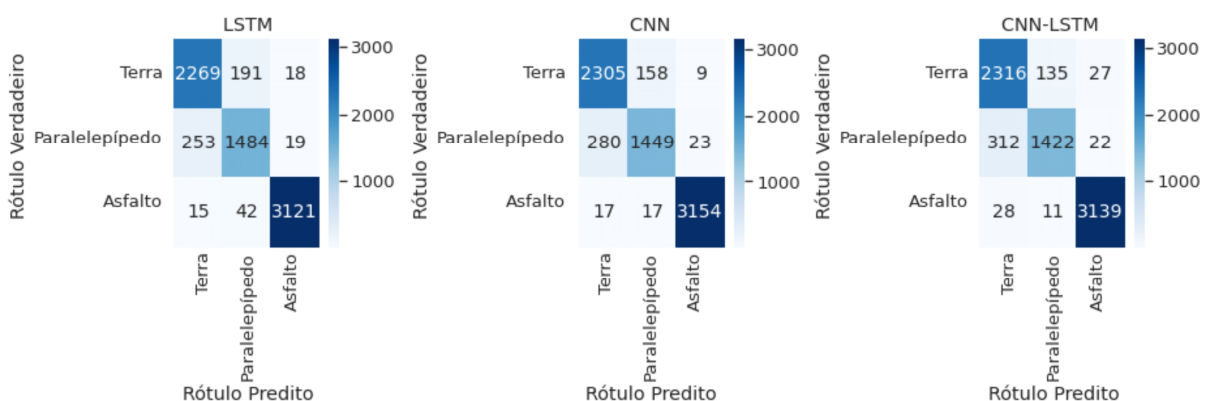
Nas redes baseadas em CNN, a utilização de duas a quatro camadas de convolução se mostrou essencial para extração das características. Na saída final dos blocos de convolução para conexão com o bloco de camadas totalmente conectadas, a utilização de *pooling* melhorou os resultados quando comparado ao uso de apenas *flattening*. Tanto a camada *Max Pooling 1D* quanto a *Global Average Pooling 1D* melhoraram os resultados, com a segunda sendo significativamente melhor. Sendo assim, todos os experimentos nos quais a média de acurácia das janelas foi maior que 90%, houve a utilização do *Global Average Pooling 1D*. O modelo baseado em CNN com os melhores resultados foi o CNN 8, com uma janela de 300 amostras, atingindo uma acurácia em treinamento de 95,56% e 93,17% em validação. Por fim, nas redes híbridas CNN-LSTM as mesmas considerações das baseadas em LSTM e CNN se aplicam, com o melhor modelo sendo a CNN-LSTM 6 com janela de 300 amostras, obtendo acurácia em treinamento de 94,9% e de 92,77% em validação. Os melhores modelos para cada abordagem de *Deep Learning* são detalhados na Tabela 27.

Tabela 27 – Valores de acurácia para os melhores modelos baseados em técnicas *deep learning*

Experimentos por Contexto					
Modelo	Fase	1	2	3	Média
LSTM	Treinamento	98,54%	98,45%	93,03%	96,67%
	Validação	93,60%	91,88%	92,70%	92,73%
CNN	Treinamento	94,39%	97,62%	94,67%	95,56%
	Validação	94,93%	92,25%	92,33%	<b>93,17%</b>
CNN-LSTM	Treinamento	91,68%	98,85%	94,16%	94,90%
	Validação	93,68%	92,67%	91,97%	92,77%

Fonte: Desenvolvido pelo autor.

Figura 35 – Matriz de confusão para cada melhor modelo de técnicas de *deep learning*



Fonte: Desenvolvido pelo autor.

Tabela 28 – Métricas de avaliação para cada melhor modelo baseado em técnicas de *deep learning*

Modelo	Classe de Dados	F1-Score	Precisão	Recall
LSTM	Asfalto	98,52%	98,83%	98,21%
	Paralelepípedo	85,46%	86,43%	84,51%
	Terra	90,49%	89,44%	91,57%
CNN	Asfalto	98,96%	99,00%	98,93%
	Paralelepípedo	85,84%	89,23%	82,71%
	Terra	90,85%	88,59%	93,24%
CNN-LSTM	Asfalto	98,62%	98,46%	98,77%
	Paralelepípedo	85,56%	90,69%	80,98%
	Terra	90,22%	87,20%	93,46%

Fonte: Desenvolvido pelo autor.

Com base na Tabela 27, é possível observar que as diferentes abordagens de *Deep Learning* se mostraram adequadas para o problema de classificação proposto. Avaliando a média de acurácia em validação, o modelo baseado em CNN foi ligeiramente melhor do que os outros, embora o baseado em LSTM foi o melhor no experimento 3, o baseado em CNN melhor no experimento 1, e o baseado em CNN-LSTM melhor no experimento 2. Analisando a matriz de confusão na Figura 35 e outras métricas de avaliação detalhadas na Tabela 28, observamos que o modelo baseado em CNN tem os maiores valores de *f1-score* para todas as classes de dados, sendo o melhor modelo quando se considera não apenas sua capacidade de obter VP e VN, mas também de evitar FP e FN. Portanto, consideramos o modelo de CNN como o melhor modelo baseado em *Deep Learning* para a classificação do tipo de superfície de pista devido a sua maior acurácia e valores de *f1-score*, além de ser o modelo com menor custo computacional entre os três modelos DNN. Sendo assim, a técnica obteve um valor de acurácia na validação de 93,17%, classificando estrada de terra com valor *f1-score* de 90,85%, paralelepípedo com 85,84% e asfalto com 98,9%, obtendo resultados melhores que os métodos clássicos e que os estudos relacionados, aplicando ainda variações contextuais.





## 7 CLASSIFICAÇÃO DE TIPO DE SUPERFÍCIE DE PISTA 2

Através dos resultados obtidos no estudo da seção anterior, foi conduzido um segundo estudo para o desenvolvimento de modelo adaptativo de classificação de tipo de superfície de pista, com foco em modelos de *Deep Learning*, os quais se mostraram mais promissores quando comparados a modelos de Aprendizado de Máquina clássicos. Neste estudo foi produzido em conjunto uma avaliação multiaspecto e multicontextual. O processo de desenvolvimento e experimentação é detalhado nas próximas subseções. Na primeira delas, o pré-processamento, os dados foram ajustados e organizados para avaliar aspectos como a influência do ponto de coleta de dados no veículo, o domínio de análise, as características de entrada do modelo e o tamanho da janela de dados. Também foi avaliada a capacidade de generalização do aprendizado dos modelos para contextos desconhecidos. Na segunda subseção, processamento, foram desenvolvidos três modelos de DNN baseados em diferentes técnicas de *Deep Learning*: LSTM, GRU e CNN. Por fim, na última subseção são detalhados e comparados os resultados obtidos.

### 7.1 PRÉ-PROCESSAMENTO

Na etapa de pré-processamento os dados foram organizados e transformados para serem utilizados como entrada dos modelos de classificação. Esta etapa incluiu a separação de dados para treinamento e validação do modelo, a definição das características de entrada, a normalização dos dados das características e o agrupamento de dados em janelas. Neste estudo, foram utilizados os dados de aceleração em três eixos (X, Y, Z) e em três pontos de coleta diferentes (abaixo e próximo da suspensão, acima e próximo da suspensão e no painel de controle), juntamente com a taxa de rotação em três eixos e nos três pontos de coleta, além da velocidade do veículo.

Com o objetivo de avaliar a influência da propriedade de dependência veicular e verificar a viabilidade de um modelo de classificação para os dados coletados em diferentes locais do veículo, definimos os seguintes *experimentos por colocação*, com base nos pontos de coleta de dados no veículo:

**Experimento por Colocação 1:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados próximo e abaixo da suspensão, mais a velocidade.

**Experimento por Colocação 2:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados próximo e acima da suspensão, mais a velocidade.

**Experimento por Colocação 3:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados no painel de controle, mais a velocidade.

Neste estudo, buscou-se identificar também o domínio de análise mais adequado para classificar o tipo de superfície de pista. Sendo assim, para cada *experimento de colocação*,

foram definidos *experimentos por domínio de análise*. Para a análise no domínio da frequência, foram aplicados nos dados a Transformada de Fourier de Curto Tempo (*Short-Time Fourier Transform* - STFT) para extrair as características de magnitude da frequência através de uma janela deslizante de 100 amostras com total sobreposição. Os *experimentos por domínio de análise* foram definidos de acordo com as características de entrada detalhadas abaixo:

**Experimento por Domínio de Análise 1:** Foram utilizadas as 7 características correspondentes a aceleração nos eixos X, Y, Z; a taxa de rotação nos eixos X, Y, Z; e a velocidade.

**Experimento por Domínio de Análise 2:** Foram utilizadas as 7 características do *Experimento por Domínio de Análise 1*, mais quatro características de aceleração composta XY, YZ, XZ, XYZ, de acordo com as fórmulas detalhadas em (TAN et al., 2019).

**Experimento por Domínio de Análise 3:** Foram utilizadas 357 características correspondentes às magnitudes das 51 frequências de cada uma das 7 características definidas no domínio do tempo em *Experimento por Domínio de Análise 1*.

**Experimento por Domínio de Análise 4:** Foram utilizadas 561 características correspondentes às magnitudes das 51 frequências de cada uma das 11 características definidas no domínio do tempo em *Experimento por Domínio de Análise 2*.

Após definir os *experimentos por domínio de análise*, os dados foram normalizados usando *Min Max Scaler*. Os dados das características no domínio do tempo foram normalizados na faixa de  $[-1,1]$ , uma vez que o sinal, neste caso, contém informações importantes, denotando a direção do vetor força. No domínio da frequência, como a magnitude não apresenta valores negativos, os dados foram normalizados na faixa de  $[0,1]$ . Em seguida, foram definidos os dados a serem usados no treinamento e na validação dos modelos. Para avaliar corretamente a generalização do aprendizado de cada técnica, avaliando sua adaptabilidade para contextos desconhecidos onde há variação nas propriedades de dependência veicular, de condução e ambiental, os dados de treinamento e validação foram divididos da mesma maneira que no estudo da seção anterior, separando-os de acordo com o conjunto de dados em três *experimentos por contexto*:

**Experimento por Contexto 1:** O modelo aprende dados de todos os veículos e motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (65%):** PVS 1, PVS 3, PVS 4, PVS 6, PVS 7, PVS 9.
- **Validação (35%):** PVS 2, PVS 5, PVS 8.

**Experimento por Contexto 2:** O modelo aprende dados de todos os cenários para alguns veículos e alguns motoristas; mas não todos os veículos com todos os motoristas para todos os cenários.

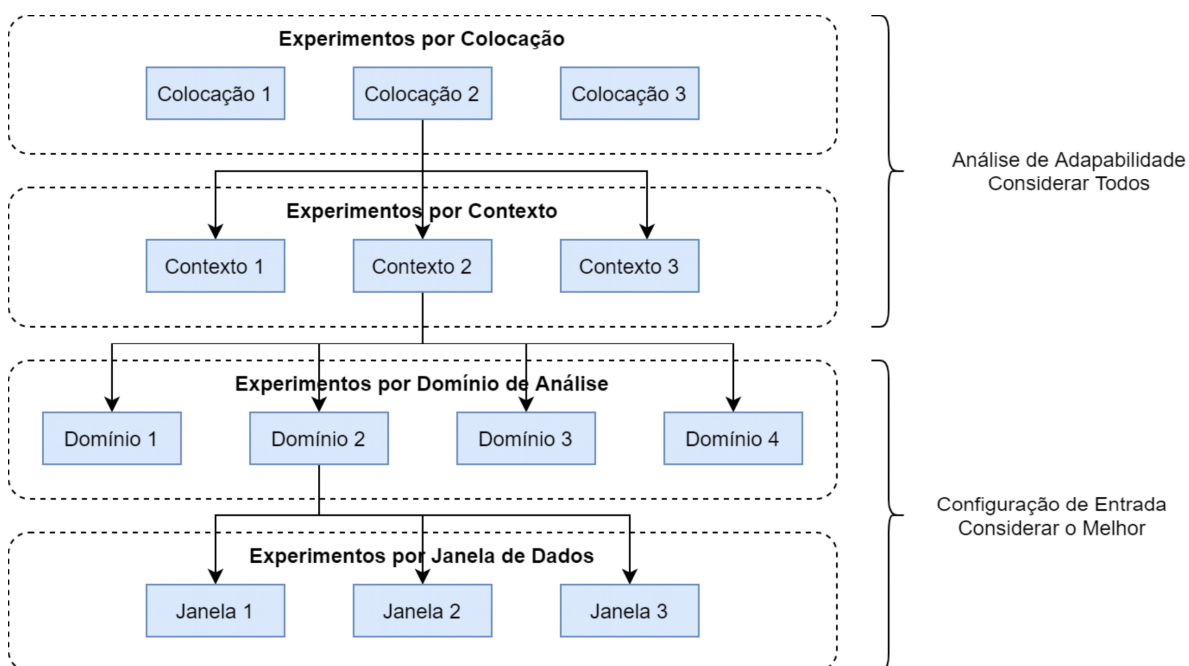
- **Treinamento (66%):** PVS 1, PVS 2, PVS 3, PVS 7, PVS 8, PVS 9.
- **Validação (34%):** PVS 4, PVS 5, PVS 6.

**Experimento por Contexto 3:** O modelo aprende dados de alguns veículos com alguns motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 4, PVS 6, PVS 8, PVS 9.
- **Validação (34%):** PVS 3, PVS 5, PVS 7.

Finalmente, para analisar a influência do número de amostras na classificação, foram criados três *experimentos por tamanho de janela de dados*, com janelas de 100, 200 e 300 amostras. A janela de dados utilizada foi fixa sem sobreposição, e cada amostra correspondeu a um vetor com os valores das características. Em resumo, cada experimento realizado neste estudo é um elemento do produto cartesiano entre *experimentos por colocação*, *experimentos por domínio de análise*, *experimentos por contexto* e *experimentos por tamanho de janela de dados*, conforme ilustra a Figura 36.

Figura 36 – Combinação de tipos de experimentos



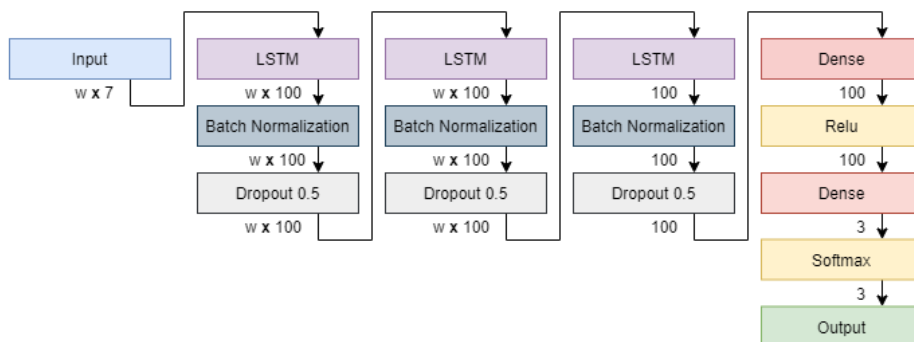
Fonte: Desenvolvido pelo autor.

## 7.2 PROCESSAMENTO

Após a etapa de pré-processamento, os dados foram aplicados em modelos de classificação baseados em *Deep Learning*. Neste estudo, foram desenvolvidas DNN baseadas em

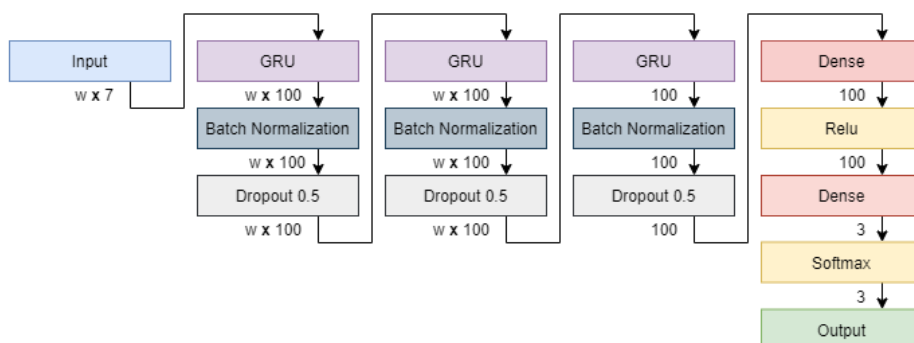
LSTM, GRU e CNN. Todos os modelos são sequenciais e utilizam o otimizador Adam em conjunto com a função de perda de Entropia Cruzada Categórica. O modelo baseado em LSTM é ilustrado na Figura 37, sendo composto por um bloco de camada de entrada, três blocos de camadas de recorrência e regularização e um bloco de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, onde *janelas* são os agrupamentos de janelas, *sequências* são os dados que compõem uma das janelas e *características* os valores das 7 variáveis. Cada bloco de recorrência e regularização é composto por uma camada LSTM unidirecional de 100 unidades, seguido por uma camada de *Batch Normalization* e outra de *Dropout* em 50%. Após o processamento nas camadas recorrentes, os parâmetros passam para o bloco de camadas totalmente conectadas, onde existem duas camadas *Dense*, a primeira com 100 neurônios com ativação *Relu* e a segunda com 3 neurônios e ativação *Softmax*, produzindo a classificação. A saída esperada da rede são os rótulos correspondentes à última amostra na janela. O modelo baseado em GRU desenvolvido possui a mesma estrutura LSTM, trocando camadas LSTM por GRU conforme detalha a Figura 38.

Figura 37 – Modelo LSTM para classificação do tipo de superfície de pista



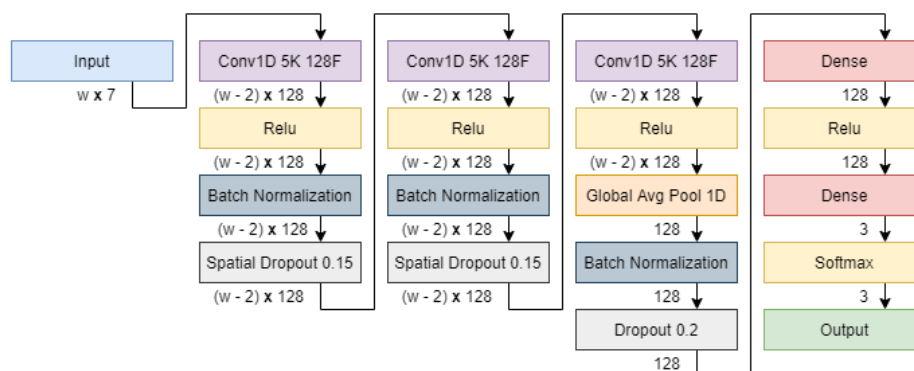
Fonte: Desenvolvido pelo autor.

Figura 38 – Modelo GRU para classificação do tipo de superfície de pista



Fonte: Desenvolvido pelo autor.

Figura 39 – Modelo CNN para classificação do tipo de superfície de pista



Fonte: Desenvolvido pelo autor.

Por fim, o modelo desenvolvido baseado em CNN é ilustrado na Figura 39, sendo composto por um bloco de camada de entrada, três blocos de camadas de convolução e regularização e um bloco de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, similar aos modelos LSTM e GRU. Os dois primeiros blocos de convolução e regularização são compostos cada um por uma camada Conv 1D com 128 filtros de *kernel* tamanho 5 para extração de características e ativação *Relu*, seguida por regularização através de uma camada de *Batch Normalization* e outra de *Spatial Dropout 1D* em 15%. O último bloco de convolução e regularização possui uma camada Conv 1D com as mesmas configurações das anteriores, uma camada de *Global Average Pooling 1D*, e camadas de regularização por *Batch Normalization* e *Dropout* em 20%. Finalmente, o bloco de totalmente conectadas consiste em duas camadas *Dense*, uma com 32 neurônios e ativação *Relu* e outra com 3 neurônios e ativação *Softmax*. A saída esperada da rede são os rótulos mais presentes na janela de dados.

Os demais hiperparâmetros das técnicas de *Deep Learning* foram utilizados com seus valores padrões, disponíveis em TensorFlow (2021d), TensorFlow (2021c) e TensorFlow (2021a).

### 7.3 ANÁLISE DE RESULTADOS

Todos os modelos desenvolvidos neste estudo foram programados em Python 3, utilizando a biblioteca Keras do Tensorflow. Os experimentos foram realizados no Google *Collaboratory*, em um ambiente com as mesmas configurações do estudo da seção anterior. Todas as configurações de treinamento estão detalhadas nos códigos-fonte documentados disponíveis na página do projeto no Github. Todos os experimentos foram executados três vezes, recuperando apenas o melhor modelo entre as três execuções. O melhor modelo foi considerado aquele que apresentou o maior valor de acurácia durante a validação. Os resultados obtidos foram agrupados por ponto de coleta de dados no veículo, detalhados nas Tabelas 29, 30 e 31. Além do

agrupamento por ponto de coleta, em cada tabela é detalhado o experimento por modelo de DNN, domínio de análise, características de entrada e janelas de dados. Cada valor exibido corresponde a média de acurácia na validação dos três *experimentos por contexto*, os quais visam analisar a capacidade de aprendizado e generalização do modelo para cenários não conhecidos, onde há variações contextuais correspondentes aos fatores de dependência veicular, de condução e ambientais. Os valores destacados correspondem ao valor máximo de média de acurácia em validação para dado tipo de DNN.

Tabela 29 – Média de acurácia para colocação próximo e abaixo da suspensão

Modelo	Domínio	Características	Tamanho da Janela de Dados		
			100	200	300
LSTM	Tempo	7	89,69%	92,06%	92,34%
		11	89,70%	90,99%	92,11%
	Frequência	357	90,34%	91,86%	92,63%
		561	90,65%	91,55%	92,82%
GRU	Tempo	7	90,23%	92,06%	92,88%
		11	89,81%	91,65%	92,31%
	Frequência	357	90,18%	91,09%	92,33%
		561	90,55%	91,59%	92,40%
CNN	Tempo	7	91,16%	92,92%	93,04%
		11	90,24%	92,58%	92,60%
	Frequência	357	89,56%	91,92%	93,91%
		561	89,78%	92,08%	93,67%

Fonte: Desenvolvido pelo autor.

Tabela 30 – Média de acurácia para colocação próximo e acima da suspensão

Modelo	Domínio	Características	Tamanho da Janela de Dados		
			100	200	300
LSTM	Tempo	7	89,23%	90,56%	90,89%
		11	87,56%	88,96%	90,58%
	Frequência	357	88,41%	89,55%	90,78%
		561	88,04%	90,00%	90,15%
GRU	Tempo	7	87,53%	89,16%	89,42%
		11	87,54%	88,61%	89,79%
	Frequência	357	88,57%	89,49%	90,41%
		561	87,84%	89,76%	90,05%
CNN	Tempo	7	88,65%	90,85%	92,02%
		11	88,25%	90,04%	90,74%
	Frequência	357	88,32%	90,75%	91,01%
		561	88,23%	90,67%	90,92%

Fonte: Desenvolvido pelo autor.

Tabela 31 – Média de acurácia para colocação no painel de controle

Modelo	Domínio	Características	Tamanho da Janela de Dados		
			100	200	300
LSTM	Tempo	7	89,64%	91,79%	91,64%
		11	88,54%	90,96%	91,73%
	Frequência	357	88,61%	89,47%	90,92%
		561	89,90%	91,03%	91,55%
GRU	Tempo	7	89,82%	91,67%	91,91%
		11	89,62%	90,83%	91,73%
	Frequência	357	88,76%	90,53%	90,74%
		561	89,67%	90,92%	91,68%
CNN	Tempo	7	89,44%	91,46%	93,05%
		11	89,64%	91,50%	93,30%
	Frequência	357	89,40%	91,51%	92,37%
		561	89,70%	92,09%	93,57%

Fonte: Desenvolvido pelo autor.

Através das tabelas de resultados, é possível observar que o tamanho da janela de dados pouco influencia o resultado final. Considerando todos os experimentos realizados, o aumento da janela de dados de 100 para 200 amostras aumentou em média cerca de 1,8% a média de acurácia em validação, enquanto que o aumento de 200 amostras para 300 acrescentou cerca de 0,8%. Em relação a utilização das variáveis brutas dos sensores (7 características) e a magnitude de suas frequências (357 características) quando comparadas a adição das variáveis de aceleração composta (11 características) e a magnitude das frequências (561 características) foi observado uma variação média de 0,1% no valor de acurácia, sendo que em 54% dos melhores experimentos não utilizaram da aceleração composta, e outros 46% utilizaram. De maneira geral, a adição destas variáveis não se justifica, aumentando o custo computacional sem ter um aumento consistente de acurácia.

Em relação aos domínios de análise, tanto o domínio do tempo quanto o domínio da frequência obtiveram bons resultados. Quando comparados os experimentos com as variáveis no domínio do tempo com sua respectiva representação no domínio da frequência, metade dos experimentos foi melhor no domínio do tempo e a outra metade teve melhor desempenho no domínio da frequência. Contudo, a variação média de acurácia entre os domínios de análise é de apenas 0,05% e, considerando o custo de transformação entre domínios e o aumento expressivo de características de entrada quando representadas no domínio da frequência, passando de 7 no domínio do tempo para 357 com sua representação na frequência, e de 11 no domínio do tempo para 561 com sua representação na frequência, o domínio do tempo mostra-se mais adequado.

Analisando os modelos de DNN propostos, observamos que todos obtiveram bons resultados, com pouca variação de resultados. Considerando todos os experimentos, a rede baseada em CNN obteve melhores resultados, seguida pela LSTM e pela GRU. Contudo, a

variação média dos valores de acurácia entre os três modelos é menor que 1%. Em relação aos diferentes pontos de coleta no veículo, observamos que as redes são capazes de classificar corretamente mesmo com interferência da dependência veicular. Em uma análise ampla, as redes têm mais facilidade em reconhecer os padrões dos dados amostrados próximo abaixo da suspensão, seguidos pelos amostrados no painel de controle e por fim os amostrados próximo e acima da suspensão. Contudo, a variação média de acurácia entre os pontos de coleta é pequena, cerca de 2%.

Em uma análise macro de todos os experimentos, observamos que as redes propostas nas suas diversas variações de configuração de dados de entrada realizam a classificação de superfície de pista com bons valores de acurácia em validação, validando a hipótese deste estudo e evidenciando a capacidade das redes de *Deep Learning* aprenderem as relações dos dados com as propriedades de dependência, generalizando para cenários desconhecidos. Com base em todas as análises individuais realizadas, consideramos o melhor modelo de classificação de pista o modelo baseado em CNN, processando no domínio do tempo 7 características correspondentes aos dados brutos dos sensores, em uma janela de dados de 300 amostras. A Tabela 32 detalha os valores de acurácia em treinamento e validação do modelo CNN para cada *experimento por contexto* e *experimento por colocação*. A Tabela 33 detalha as demais métricas de avaliação e a Figura 40 detalha as matrizes de confusão correspondentes.

Tabela 32 – Valores de acurácia para o modelo baseado em CNN

Experimento por Colocação	Fase de Processamento	Experimento por Contexto			
		1	2	3	Média
Próximo e abaixo da suspensão	Treinamento	94,67%	94,40%	97,01%	95,36%
	Validação	94,78%	92,17%	92,17%	93,04%
Próximo e acima da suspensão	Treinamento	88,46%	92,89%	95,45%	92,27%
	Validação	94,11%	90,14%	91,80%	92,02%
Painel de Controle	Treinamento	93,16%	94,84%	97,50%	95,17%
	Validação	95,25%	89,93%	93,96%	93,05%

Fonte: Desenvolvido pelo autor.

Através das Tabelas 32 e 33, e da Figura 40 observamos que a rede CNN demonstrou boa capacidade de aprendizado e generalização para todos os pontos de coleta e todas as variações de contexto. Para os dados amostrados próximo e abaixo da suspensão, a rede obteve média de acurácia de treinamento de 95,36% e de 93,04% em validação, classificando segmentos de asfalto com *f1-score* de 98,60%, paralelepípedo com 86,09% e terra com 90,78%. Para os dados amostrados próximo e acima da suspensão, o modelo obteve média de acurácia em treinamento de 92,27% e de 92,02% em validação, classificando segmentos de asfalto com *f1-score* de 98,76%, paralelepípedo com 84,18% e terra com 86,06%. Por fim, para os dados amostrados no painel de controle obteve-se média de acurácia em treinamento de 95,17% e de 93,05% em validação, classificando segmentos de asfalto com *f1-score* de 98,62%, paralelepípedo com 85,47% e terra com 91,12%. Observa-se que os resultados obtidos são superiores aos

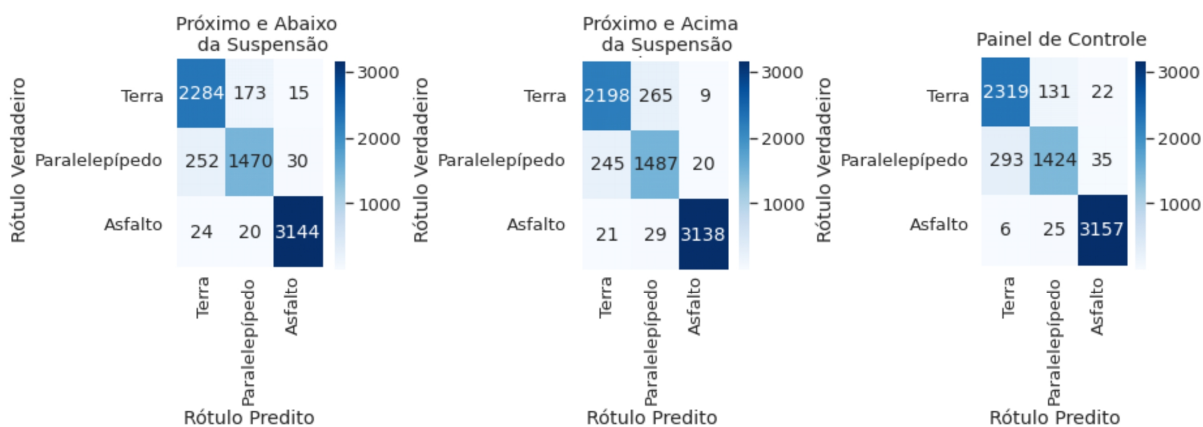


Tabela 33 – Métricas de avaliação para o modelo baseado em CNN

Colocação	Classe de Dados	F1-Score	Precisão	Recall
Próximo e abaixo da suspensão	Asfalto	98,60%	98,59%	98,62%
	Paralelepípedo	86,09%	88,39%	83,90%
	Terra	90,78%	89,22%	92,39%
Próximo e acima da suspensão	Asfalto	98,76%	99,08%	98,43%
	Paralelepípedo	84,18%	83,49%	84,87%
	Terra	89,06%	89,20%	88,91%
Painel de Controle	Asfalto	98,62%	98,23%	99,02%
	Paralelepípedo	85,47%	90,13%	81,27%
	Terra	91,12%	88,58%	93,81%

Fonte: Desenvolvido pelo autor.

Figura 40 – Matriz de confusão para o modelo de CNN em cada ponto de coleta de dados



Fonte: Desenvolvido pelo autor.

estudos anteriores, adicionando-se que neste estudo a análise foi realizada em contextos com variações das propriedades de dependência.



## 8 CLASSIFICAÇÃO DE QUALIDADE DE SUPERFÍCIE DE PISTA

Nesta seção é apresentado o estudo, desenvolvimento e experimentação voltado ao desenvolvimento de um modelo adaptativo para classificação da qualidade de superfície da pista. Baseado nos resultados dos experimentos das seções anteriores, este estudo focou no desenvolvimento de modelos baseados em *Deep Learning*. O processo de desenvolvimento e experimentação é detalhado nas próximas subseções. Na primeira delas, de pré-processamento, foi realizada a seleção de variáveis, padronizados seus dados através de normalização dos sinais, e definidos experimentos para avaliar aspectos como ponto de coleta de dados no veículo, tamanho da janela de dados e variações contextuais, onde foi possível analisar a capacidade de generalização do aprendizado dos modelos para contextos desconhecidos e, portanto, avaliar sua adaptabilidade. Na segunda subseção, de processamento, foram desenvolvidos cinco modelos de DNN baseados em diferentes técnicas de *Deep Learning*: LSTM, GRU, CNN, CNN-LSTM e ConvLSTM. Por fim, na última subseção são detalhados e comparados os resultados obtidos.

### 8.1 PRÉ-PROCESSAMENTO

Com base nos resultados obtidos nos experimentos anteriores, neste estudo utilizamos como características de entrada os dados brutos das variáveis de aceleração em três eixos (X, Y, Z) e em três diferentes pontos de coleta (abaixo e próximo da suspensão, acima e próximo da suspensão e painel de controle), juntamente com os dados de taxa rotação em três eixos e em três diferentes pontos de coleta, além da velocidade do veículo. Inicialmente, todos os dados foram normalizados com *Robust Scaler*, o qual utilizando intervalo interquartil se mostra um escalador mais robusto para dados com *outliers* (VAITHEESHWARI; SATHIESHKUMAR, 2019). Os dados das características foram normalizados mantendo o sinal, uma vez que o sinal contém informação importante de direção do vetor de força. Após a normalização dos dados, para avaliar a influência da propriedade de dependência veicular e verificar a viabilidade de um modelo de classificação para os dados coletados em diferentes pontos do veículo, foram definidos os seguintes *experimentos por colocação*, com base nos pontos de coleta de dados no veículo:

**Experimento por Colocação 1:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados próximo e abaixo da suspensão, mais a velocidade.

**Experimento por Colocação 2:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados próximo e acima da suspensão, mais a velocidade.

**Experimento por Colocação 3:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados no painel de controle, mais a velocidade.

Para avaliar a capacidade de generalização do aprendizado de cada técnica, avaliando sua adaptabilidade para contextos desconhecidos onde há variação nas propriedades de depen-

dência veicular, de condução e ambiental, os dados de treinamento e validação foram divididos da mesma maneira que as pesquisas das seções anteriores, separando-os de acordo com o conjunto de dados em três *experimentos por contexto*:

**Experimento por Contexto 1:** O modelo aprende dados de todos os veículos e motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (65%):** PVS 1, PVS 3, PVS 4, PVS 6, PVS 7, PVS 9.
- **Validação (35%):** PVS 2, PVS 5, PVS 8.

**Experimento por Contexto 2:** O modelo aprende dados de todos os cenários para alguns veículos e alguns motoristas; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 3, PVS 7, PVS 8, PVS 9.
- **Validação (34%):** PVS 4, PVS 5, PVS 6.

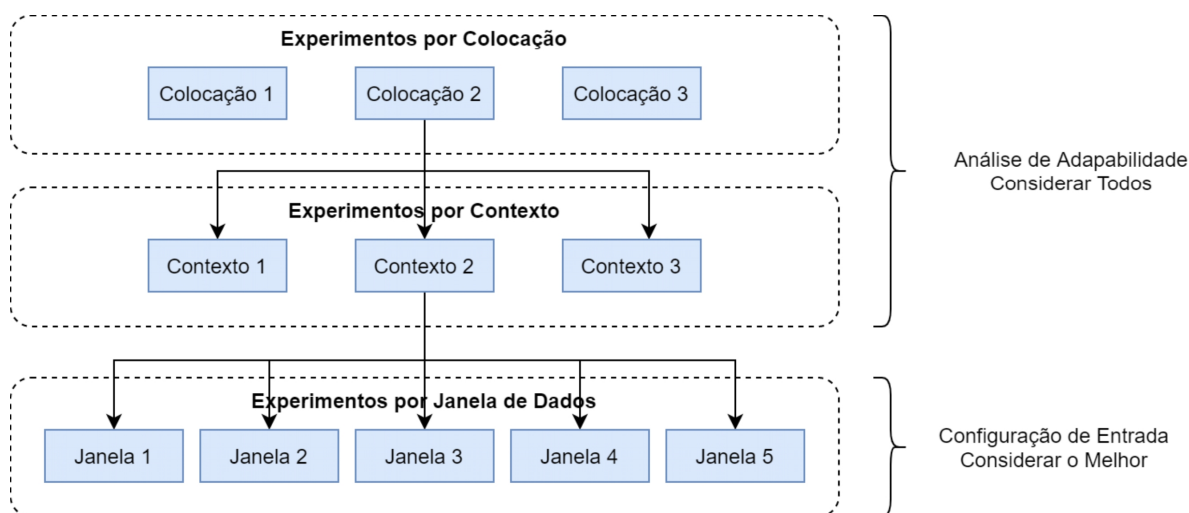
**Experimento por Contexto 3:** O modelo aprende dados de alguns veículos com alguns motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 4, PVS 6, PVS 8, PVS 9.
- **Validação (34%):** PVS 3, PVS 5, PVS 7.

Finalmente, para analisar a influência do número de amostras na classificação, foram criados três *experimentos por tamanho de janela de dados*, com janelas de 100, 200, 300, 400, e 500 amostras. A janela de dados utilizada foi fixa sem sobreposição, e cada amostra correspondeu a um vetor com os valores das características. Em resumo, cada experimento realizado neste estudo é um elemento do produto cartesiano entre *experimentos por colocação*, *experimentos por contexto* e *experimentos por tamanho de janela de dados*, conforme ilustra a Figura 41.

Para avaliar a necessidade de aplicar técnicas de balanceamento de classe de dados, foi medida a distribuição de cada classe conforme detalhado na Tabela 34. A distribuição de classes foi calculada com os dados de treinamento (HE; MA, 2013; KUHN; JOHNSON et al., 2013), uma vez que estes são os dados utilizados pelos modelos para aprender os padrões. A distribuição foi calculada em relação ao número de amostras, uma vez que as janelas de dados são definidas de acordo com esse parâmetro. Analisando a Tabela 34, observamos que a distribuição das classes de qualidade de superfície de pista situa-se entre 10,58% e 47,73%, dependendo do experimento. A proporção varia entre 1:1,1 a 1:8,5, onde para 1 amostra de uma determinada classe de dados, existem de 1,1 a 8,5 amostras nas outras classes. O desbalanceamento de classes pode ser considerado leve ou severo, onde as proporções de distribuição

Figura 41 – Combinação de tipos de experimentos



Fonte: Desenvolvido pelo autor.

que variam de 1:4 até 1:100 (presença de 20% - 1%) são consideradas desbalanceamento leve e proporções de distribuição que variam de 1:100 ou mais (<1% de presença) são consideradas desbalanceamentos severos (KRAWCZYK, 2016; BROWNLEE, 2020). Como podemos observar, as proporções de distribuição das classes de dados neste estudo em sua maioria não são classificadas sequer como desbalanceamento leve, e os poucos que são desbalanceamento leve, não há tendência a desbalanceamento severo. Sendo assim, este estudo não necessita de aplicação de técnica para balanceamento de classes. Com a definição dos experimentos, convém ressaltar a diferença entre o modelo de GT de anotação automatizada por máquina e o modelo que se busca obter com estes experimentos. No primeiro, os três modelos de KMC gerados estão cada um deles em função de um dos veículos. Já neste estudo, os experimentos buscam por um único modelo que se adapte aos três veículos.

Tabela 34 – Distribuição de classes de dados de qualidade de superfície de pista

Fonte de Dados	Classe de Dados					
	Bom		Regular		Ruim	
	Percentual	Proporção	Percentual	Proporção	Percentual	Proporção
Exp. por Contexto 1	41,69%	1:1,4	47,73%	1:1,1	10,58%	1:8,5
Exp. por Contexto 2	43,53%	1:1,3	40,17%	1:1,5	16,30%	1:5,1
Exp. por Contexto 3	42,51%	1:1,4	40,64%	1:1,5	16,85%	1:4,9

Fonte: Desenvolvido pelo autor.

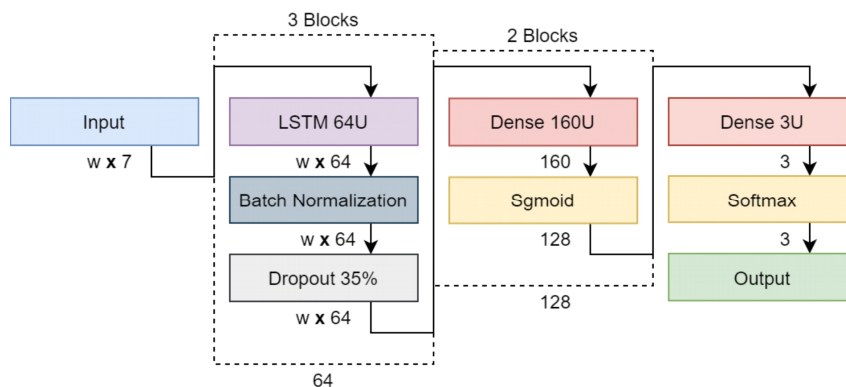
## 8.2 PROCESSAMENTO

Após o pré-processamento, os dados foram aplicados em cinco modelos de *Deep Learning*, sendo baseados em LSTM, GRU, CNN, CNN-LSTM e ConvLSTM. Todos os modelos

desenvolvidos são sequenciais e utilizam o otimizador Adam em conjunto com a função de perda Entropia Cruzada Categórica. Todos os modelos tiveram seus hiperparâmetros ajustados (*hyperparameter tuning*) utilizando o algoritmo *HyperBand* implementado na biblioteca Keras Tuner, onde foram testados diferentes tipos de camadas, número de camadas e neurônios, funções de ativação e várias outras características para encontrar o conjunto ideal de hiperparâmetros para cada DNN.

O melhor modelo baseado em LSTM obtido através do ajuste de hiperparâmetros é detalhado na Figura 42. O modelo DNN é composto de um bloco de camada de entrada, três blocos de camadas de recorrência e regularização e três blocos de camada totalmente conectadas para produção da saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, onde *janelas* são os agrupamentos de todas as janelas de dados, *sequências* são as sequências de dados para cada janela, e *características* são os valores das variáveis de entrada. Cada bloco de recorrência e regularização é composto por uma camada LSTM unidirecional de 64 unidades, seguida por uma camada de *Batch Normalization* e uma camada de *Dropout* em 35%. Após o processamento nas camadas recorrentes, os parâmetros são passados para os blocos totalmente conectados, onde existem duas camadas *Dense* com 160 neurônios e ativação *Sigmoid*, e uma camada *Dense* com 3 neurônios e ativação *Softmax*, produzindo a classificação. A saída esperada são os rótulos mais presentes na janela de dados.

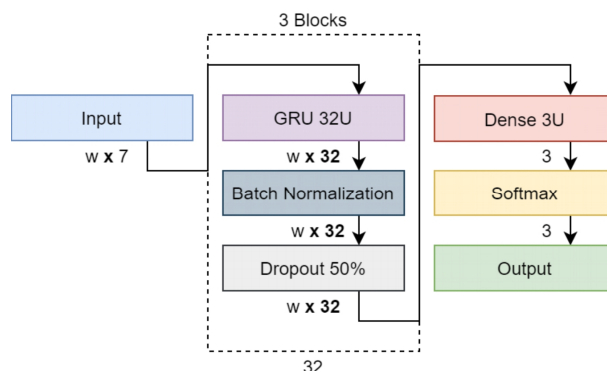
Figura 42 – Modelo LSTM para classificação de qualidade de superfície de pista



Fonte: Desenvolvido pelo autor.

O melhor modelo baseado em GRU obtido neste estudo é detalhado na Figura 43. O modelo de DNN é composto de um bloco de camada de entrada, três blocos de camadas de recorrência e regularização e um bloco de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, semelhante ao baseado em LSTM. Cada bloco de recorrência e regularização é composto por uma camada GRU unidirecional de 32 unidades, seguida por uma camada de *Batch Normalization* e uma camada de *Dropout* em 50%. Após o processamento nas camadas recorrentes, os parâmetros passam para um bloco totalmente conectado, onde existe uma camada *Dense* com 3 neurônios e ativação *Softmax*, produzindo a classificação.

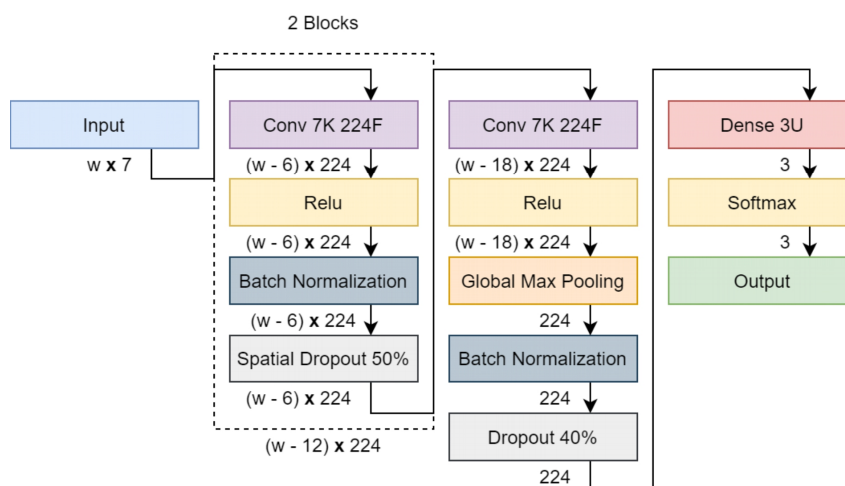
Figura 43 – Modelo GRU para classificação de qualidade de superfície de pista



Fonte: Desenvolvido pelo autor.

O melhor modelo baseado em CNN obtido é detalhado na Figura 44. O modelo de DNN é composto de um bloco de entrada, dois blocos de convolução e regularização e um bloco de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, semelhante aos modelos baseados em LSTM e GRU. O primeiro bloco de convolução e regularização é composto por uma camada Conv 1D com 224 filtros com *kernel* de tamanho 7 para extração de características e ativação *Relu*, seguida de regularização através de uma camada de *Batch Normalization* e uma camada de *Spatial Dropout 1D* em 50%. O último bloco de convolução e regularização tem uma camada Conv 1D com as mesmas configurações do anterior, uma camada *Global Max Pooling 1D* para extrair recursos mais robustos por meio dos valores máximos em cada região, e camadas de regularização por *Batch Normalization 1D* e *Dropout* em 40%. Finalmente, o bloco totalmente conectado consiste de uma camada *Dense* com 3 neurônios e ativação *Softmax*, produzindo a classificação.

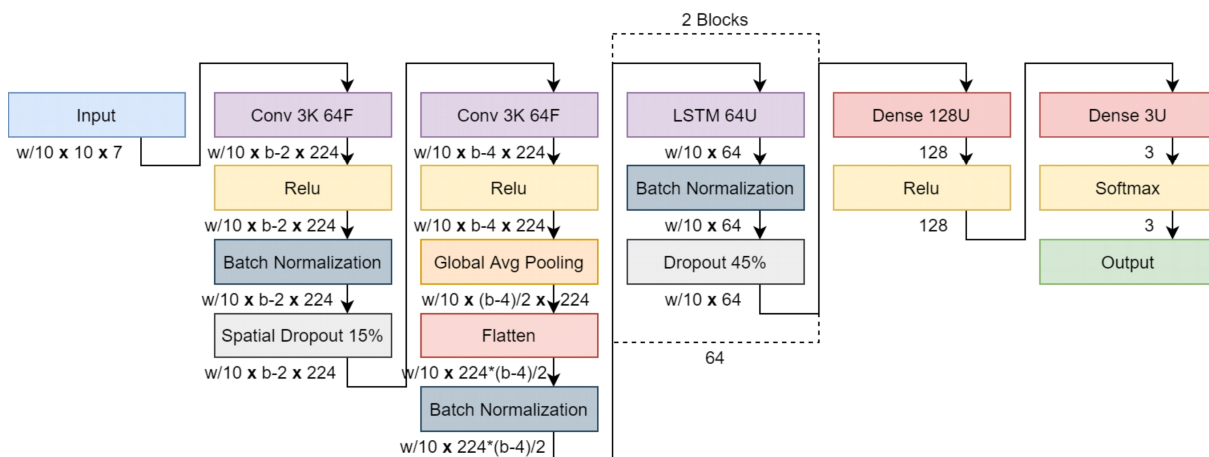
Figura 44 – Modelo CNN para classificação de qualidade de superfície de pista



Fonte: Desenvolvido pelo autor.

O melhor modelo híbrido baseado em CNN-LSTM obtido é detalhado na Figura 45. O modelo de DNN é composto de um bloco de entrada, dois blocos de convolução e regularização, dois blocos recorrentes e de regularização e dois blocos de camada totalmente conectadas para produção de saída. O bloco de entrada tem uma camada que recebe um tensor *janelas x sequências x subseqüências x características*, onde *janelas* são os agrupamentos de todas as janelas de dados, *sequências* são as sequências de dados para cada janela, *subseqüências* são as subpartes da sequência de dados original, e *características* os valores das variáveis de entrada. O primeiro bloco de convolução e regularização é composto por uma camada Conv 1D com 64 filtros com *kernel* de tamanho 3 para extração de características e ativação *Relu*, seguida de regularização através de uma camada de *Batch Normalization* e uma camada de *Spatial Dropout 1D* em 15%. O último bloco de convolução e regularização tem uma camada Conv 1D com as mesmas configurações do anterior, uma camada *Global Average Pooling 1D* para extrair recursos mais robustos por meio da média dos valores em cada região, uma camada *Flatten* para reagrupar os recursos extraídos na sequência temporal original, e regularização por camada de *Batch Normalization*. Em seguida, cada bloco de recorrência e regularização é composto por uma camada LSTM unidirecional de 64 unidades, seguida por uma camada de *Batch Normalization* e uma camada de *Dropout* em 45%. Finalmente, os parâmetros resultantes passam para duas camadas *Dense*, a primeira com 128 neurônios e ativação de *Relu*, e a segunda com 3 neurônios e ativação de *Softmax*, produzindo a classificação.

Figura 45 – Modelo CNN-LSTM para classificação de qualidade de superfície de pista



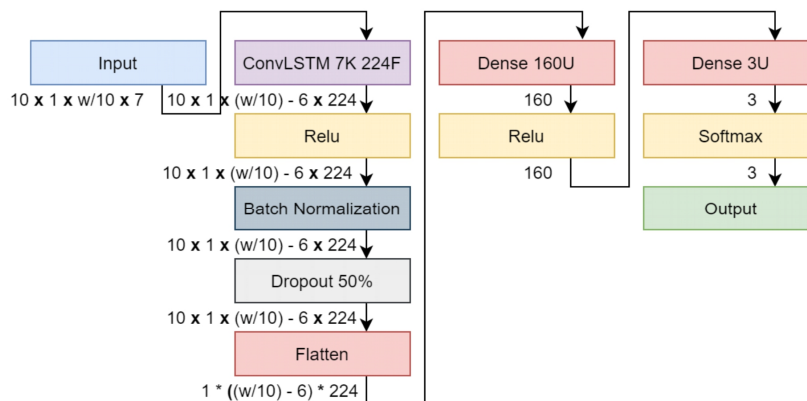
Fonte: Desenvolvido pelo autor.

O melhor modelo baseado em ConvLSTM obtido é detalhado na Figura 46. O modelo de DNN é composto de um bloco de entrada, um bloco convolucional recorrente e de regularização, e dois blocos de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x subseqüências x características*, semelhante ao modelo CNN-LSTM. O bloco convolucional recorrente e de regularização é composto por uma camada *ConvLSTM 1D* com 224 filtros *kernel* de tamanho 7 e ativação *Relu*, seguido por regularização através de uma camada *Dropout* em 35%. Finalmente,



os parâmetros resultantes são achatados (*flattened*) e passados para duas camadas *Dense*, a primeira com 160 neurônios e ativação *Relu*, e a segunda com 3 neurônios e ativação *Softmax*, produzindo a classificação.

Figura 46 – Modelo ConvLSTM para classificação de qualidade de superfície de pista



Fonte: Desenvolvido pelo autor.

Os demais hiperparâmetros das técnicas de *Deep Learning* foram utilizados com seus valores padrões, disponíveis em TensorFlow (2021d), TensorFlow (2021c), TensorFlow (2021a) e TensorFlow (2021b).

### 8.3 ANÁLISE DE RESULTADOS

Neste estudo todos os modelos de classificação de qualidade de superfície foram desenvolvidos na linguagem Python 3, utilizando da biblioteca Keras 2, a qual é uma API de alto nível do TensorFlow. Os hiperparâmetros dos modelos foram afinados através da biblioteca Keras Tuner. Todos os experimentos foram executados em máquinas do Google *Collaboratory*, com as mesmas configurações dos estudos das seções anteriores. Todas as configurações de treinamento estão detalhadas nos códigos-fonte documentados disponíveis na página do projeto no Github. Cada experimento executado é um elemento do produto Cartesiano entre *experimentos por colocação*, *experimentos por contexto* e *experimentos por tamanho da janela de dados*. Cada experimento foi executado três vezes, para minimizar a aleatoriedade de parâmetros iniciais, como pesos sinápticos, recuperando-se o melhor das três execuções. Consideramos a melhor execução aquela com maior valor de acurácia na fase de validação, uma vez que o treinamento dos modelos foi configurado para maximizar a métrica de acurácia.

Os resultados obtidos com a execução de todos os experimentos são apresentados nas Tabelas 35 a 39. Em cada tabela são detalhados os resultados para determinado modelo de DNN, para cada colocação, contexto e janela de dados. Também são apresentadas métricas de média aritmética para cada tipo de experimento, sendo Acurácia Média dos Experimentos por Contexto (AMEC) e Acurácia Média dos Experimentos por Contexto e Colocação (AMECC), onde são destacados os menores e maiores valores de acurácia obtidos na fase de validação.

Tabela 35 – Valores de acurácia em validação obtidos pelo modelo LSTM

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	89,79%	90,89%	91,40%	91,67%	91,61%
	2	90,37%	91,31%	93,00%	92,49%	90,94%
	3	91,42%	92,20%	92,90%	92,00%	92,99%
	AMEC	90,53%	91,47%	92,43%	92,05%	91,85%
Próximo e Acima da Suspensão	1	90,14%	91,10%	92,42%	92,03%	91,55%
	2	90,63%	92,11%	92,34%	92,27%	91,42%
	3	91,54%	92,26%	92,50%	93,04%	91,56%
	AMEC	90,77%	91,82%	92,42%	92,44%	91,51%
Painel de Controle	1	90,45%	90,60%	92,50%	89,83%	88,86%
	2	87,96%	88,36%	91,55%	90,39%	90,11%
	3	91,46%	91,68%	92,09%	92,71%	92,31%
	AMEC	89,95%	90,21%	92,05%	90,98%	90,43%
	AMECC	90,42%	91,17%	92,30%	91,82%	91,26%

Fonte: Desenvolvido pelo autor.

Tabela 36 – Valores de acurácia em validação obtidos pelo modelo GRU

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	90,27%	91,20%	92,26%	91,61%	92,73%
	2	90,89%	92,08%	92,75%	92,21%	91,98%
	3	91,47%	92,34%	92,99%	92,87%	92,99%
	AMEC	90,88%	91,87%	92,67%	92,23%	92,57%
Próximo e Acima da Suspensão	1	90,31%	92,38%	92,77%	92,82%	93,38%
	2	90,40%	92,00%	93,41%	93,04%	92,25%
	3	91,84%	92,83%	93,15%	92,93%	93,20%
	AMEC	90,85%	92,40%	93,11%	92,93%	92,94%
Painel de Controle	1	91,14%	91,91%	93,60%	92,66%	92,92%
	2	90,56%	89,85%	87,74%	88,07%	86,93%
	3	91,65%	92,69%	93,76%	94,67%	93,06%
	AMEC	91,12%	91,48%	91,70%	91,80%	90,97%
	AMECC	90,95%	91,92%	92,49%	92,32%	92,16%

Fonte: Desenvolvido pelo autor.

Tabela 37 – Valores de acurácia em validação obtidos pelo modelo CNN

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	89,86%	91,65%	92,50%	92,66%	92,66%
	2	90,09%	91,17%	93,50%	92,82%	93,50%
	3	92,16%	92,88%	94,09%	94,18%	94,69%
	AMEC	90,70%	91,90%	93,36%	93,22%	93,62%
Próximo e Acima da Suspensão	1	90,75%	92,75%	93,75%	94,29%	93,91%
	2	91,31%	92,60%	93,91%	93,98%	93,43%
	3	92,05%	93,37%	93,80%	94,40%	94,35%
	AMEC	91,37%	92,91%	93,82%	94,22%	93,90%
Painel de Controle	1	90,90%	92,83%	94,78%	94,92%	95,35%
	2	90,88%	89,16%	89,85%	89,01%	88,59%
	3	92,26%	93,51%	94,49%	95,10%	95,17%
	AMEC	91,35%	91,83%	93,04%	93,01%	93,04%
	AMECC	91,14%	92,21%	93,41%	93,48%	93,52%

Fonte: Desenvolvido pelo autor.

Tabela 38 – Valores de acurácia em validação obtidos pelo modelo CNN-LSTM

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	90,00%	91,10%	91,28%	91,09%	91,61%
	2	90,31%	90,78%	92,38%	92,87%	91,49%
	3	91,85%	92,15%	92,86%	92,44%	93,06%
	AMEC	90,72%	91,34%	92,17%	92,13%	92,06%
Próximo e Acima da Suspensão	1	90,47%	91,05%	93,40%	91,88%	92,14%
	2	90,56%	91,97%	92,67%	93,76%	92,12%
	3	91,84%	92,72%	92,74%	93,09%	92,79%
	AMEC	90,96%	91,91%	92,94%	92,91%	92,35%
Painel de Controle	1	90,89%	91,99%	92,62%	92,45%	92,66%
	2	89,45%	88,63%	89,11%	89,01%	88,73%
	3	91,85%	92,55%	92,90%	92,71%	92,31%
	AMEC	90,73%	91,06%	91,54%	91,39%	91,23%
	AMECC	90,80%	91,44%	92,22%	92,14%	91,88%

Fonte: Desenvolvido pelo autor.

Tabela 39 – Valores de acurácia em validação obtidos pelo modelo ConvLSTM

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	88,91%	90,05%	89,63%	90,51%	86,63%
	2	90,00%	90,56%	90,56%	91,71%	90,94%
	3	90,89%	91,63%	91,52%	91,57%	90,68%
	AMEC	89,93%	90,75%	90,57%	91,26%	89,42%
Próximo e Acima da Suspensão	1	88,74%	89,58%	90,77%	87,53%	79,29%
	2	90,31%	90,67%	90,35%	90,88%	89,28%
	3	90,84%	91,41%	89,89%	88,30%	89,05%
	AMEC	89,97%	90,56%	90,33%	88,90%	85,87%
Painel de Controle	1	87,74%	89,79%	89,59%	81,66%	81,52%
	2	89,87%	90,18%	88,15%	85,80%	84,72%
	3	90,94%	91,30%	90,09%	89,93%	90,68%
	AMEC	89,52%	90,42%	89,28%	85,80%	85,64%
	AMECC	89,81%	90,58%	90,06%	88,66%	86,98%

Fonte: Desenvolvido pelo autor.

Em nossa análise, para avaliar a habilidade dos modelos generalizarem seu aprendizado para contextos desconhecidos, como diferentes veículos, motoristas ou ambientes, consideramos que o modelo deve obter bom desempenho nos três *experimentos por contexto*. Sendo assim, nossa análise é pautada inicialmente na média de acurácia destes experimentos, representada pela métrica AMEC. Através dos valores AMEC detalhados nas Tabelas 35 a 39, observamos que todos os modelos de DNN desenvolvidos obtiveram bons resultados independentes do contexto, com o valor de média de acurácia variando entre 85,64% até 94,22% conforme modelo e ponto de coleta de dados.

Analisando o impacto do tamanho das janelas de dados, observamos os modelos LSTM, GRU, CNN e CNN-LSTM possuem comportamento semelhante, enquanto que o modelo ConvLSTM se diferencia completamente dos demais. Para os quatro primeiros modelos, as janelas de 100 e 200 amostras obtiveram todos os piores resultados na grande maioria dos experimentos e modelos, denotando ser uma quantidade de amostras insuficientes para classificar a qualidade da pista com uma boa confiabilidade. Nestas duas janelas estão as maiores variações dos valores de acurácia, onde a janela de dados de 100 amostras obteve acurácia média de 90,83%, e a de 200 amostras obteve 91,69%. Por outro lado, as janelas de 300, 400 e 500 amostras obtiveram os melhores resultados, com uma estabilização dos valores de acurácia, apresentando uma variação muito pequena. Na média entre todos os experimentos dos quatro modelos, a janela de 300 amostras obteve acurácia de 92,61%, a de 400 obteve 92,44%, e a de 500 obteve 92,21%. Já para o modelo ConvLSTM, os experimentos com janelas de 500 amostras resultaram nos piores resultados, seguidos pela janela de 400 e a de 100. As janelas de 200 e 300 amostras produziram os melhores resultados, embora consideravelmente piores que os demais modelos.

Em relação aos pontos de coleta de dados, todos obtiveram bons resultados, com pe-

quena variação de uma colocação para outra. Os valores de média acurácia AMEC entre os modelos para a colocação próximo e abaixo da suspensão variaram de 89,42% até 93,62%; para próximo e acima da suspensão de 85,87% até 94,22%; e no painel de controle de 85,64% até 93,04%. Até este ponto, o modelo baseado em CNN se mostrou o melhor para todas as colocações, mas com janela de dados de 500 amostras para sensores empregados próximo e abaixo da suspensão; janela de 400 amostras para dados próximo e acima da suspensão; e 300 amostras para sensores empregados no painel de controle. Neste estudo consideramos que o melhor modelo deve possibilitar sua operação independentemente do local de colocação dos sensores no veículo. Portanto, deve ser aquele com melhor desempenho entre os diferentes pontos de coleta para uma dada janela de dados, representado pela média de acurácia entre os *experimentos por colocação*. Sendo assim, esta métrica é detalhada nas tabelas por AMECC, onde para determinar a melhor configuração de janela para cada modelo é considerado a média de acurácia entre todos os *experimentos por colocação* e *experimentos por contexto*.

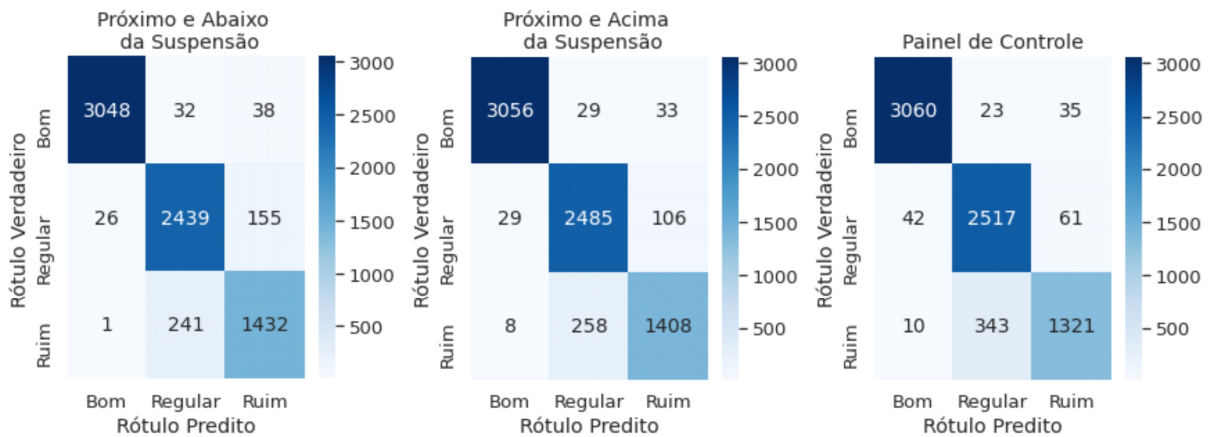
Através da métrica supracitada, observamos que o melhor modelo é o baseado em CNN em janela de 500 amostras, resultando em média de acurácia de 93,52%. O segundo melhor é o baseado em GRU com 300 amostras resultando em 92,49%. O terceiro o baseado em LSTM de 300 amostras com 92,30%. Em quarto o baseado em CNN-LSTM de 300 amostras com 92,22%. Por último, o modelo baseado em ConvLSTM de 200 amostras resultando em 90,58%. A melhor configuração de cada modelo é detalhada na Tabela 40 com as demais métricas de avaliação. Todos os valores das métricas apresentadas nesta tabela correspondem a média entre os *experimentos por colocação* e *experimentos por contexto*.

Tabela 40 – Métricas de avaliação para a melhor configuração de cada modelo DNN

Métrica de Avaliação	Modelo	LSTM	GRU	CNN	CNN-LSTM	ConvLSTM
	Janela	300	300	500	300	200
Acurácia	Treinamento	93,58%	93,44%	91,70%	92,28%	94,48%
	Validação	92,30%	92,49%	93,52%	92,22%	90,58%
Precisão	Boa	98,25%	98,30%	98,98%	98,76%	97,82%
	Regular	88,65%	88,66%	89,20%	87,43%	86,41%
	Ruim	85,32%	87,81%	90,11%	87,53%	82,50%
	Média	90,74%	91,59%	92,76%	91,24%	88,91%
Recall	Boa	97,23%	97,90%	98,04%	97,47%	97,48%
	Regular	90,97%	91,77%	94,62%	92,25%	87,28%
	Ruim	84,08%	81,11%	81,89%	79,25%	80,81%
	Média	90,76%	90,26%	91,52%	89,66%	88,52%
F1-Score	Boa	97,73%	98,10%	98,51%	98,11%	97,64%
	Regular	89,75%	90,09%	91,75%	89,69%	86,78%
	Ruim	84,63%	83,62%	85,16%	82,58%	81,27%
	Média	90,70%	90,60%	91,81%	90,13%	88,56%

Fonte: Desenvolvido pelo autor.

Figura 47 – Matriz de confusão para o modelo CNN em cada ponto de coleta de dados



Fonte: Desenvolvido pelo autor.

Como podemos observar na Tabela 40, todas as classes de dados foram corretamente tratadas por todos os modelos sem haver viés, com valores acima de 80%. Neste estudo, consideramos que o melhor modelo deve ser aquele que maximize tanto os VP quanto os VN, e minimize os FN e FP, de forma que a métrica *f1-score* associa estes dois fatores. Analisando os valores, observamos que a rede CNN obteve os melhores resultados para praticamente todas as métricas em todas as classes, obtendo o melhor *f1-score* em todas as classes. Sendo assim, consideramos o modelo baseado em CNN com janela de 500 amostras o melhor modelo para classificação de qualidade de superfície de pista, com valor de média entre *experimentos por colocação* e *experimentos por contexto* de 93,52% para acurácia, 92,76% para precisão, 91,52% para *recall* e 91,91% para *f1-score*. Na Figura 47 é ilustrado a matriz confusão para o modelo baseado em CNN em cada ponto de coleta de dados.

## 9 RECONHECIMENTO DE LOMBADAS

Nesta seção é apresentado o estudo voltado ao desenvolvimento de um modelo adaptativo para detecção de lombadas na via. Baseado nos experimentos das seções anteriores, este estudo focou no desenvolvimento de modelos baseados em *Deep Learning*. Contudo, nos anteriores foi produzido modelos para classificação multiclasse de eventos persistentes, enquanto neste os modelos consistem de uma classificação binária de evento transiente. O processo de desenvolvimento e experimentação é detalhado nas próximas subseções. Na primeira delas, o pré-processamento, foi realizada a seleção de variáveis, padronizado seus dados através de normalização dos sinais, e definido experimentos para avaliar aspectos como ponto de coleta de dados no veículo, tamanho da janela de dados e variações contextuais, onde foi possível analisar a capacidade de generalização do aprendizado dos modelos para contextos desconhecidos e, portanto, avaliar sua adaptabilidade. Na segunda subseção, processamento, foram desenvolvidos cinco modelos de DNN baseados em diferentes técnicas de *Deep Learning*: LSTM, GRU, CNN, CNN-LSTM e ConvLSTM. Por fim, na última subseção são detalhados e comparados os resultados obtidos.

### 9.1 PRÉ-PROCESSAMENTO

Com base nos resultados obtidos nos experimentos das seções anteriores, neste estudo utilizamos como características de entrada os dados brutos das variáveis de aceleração em três eixos (X, Y, Z) e em três diferentes pontos de coleta (abaixo e próximo da suspensão, acima e próximo da suspensão e painel de controle), juntamente com os dados de taxa rotação em três eixos e em três diferentes pontos de coleta, além da velocidade do veículo. Inicialmente, todos os dados foram normalizados com *Robust Scaler*, mantendo o sinal de direção do vetor de força. Após a normalização dos dados, para avaliar a influência da propriedade de dependência veicular e verificar a viabilidade de um modelo de classificação para os dados coletados em diferentes pontos do veículo, foram definidos os seguintes *experimentos por colocação*, com base nos pontos de coleta de dados no veículo:

**Experimento por Colocação 1:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados próximo e abaixo da suspensão, mais a velocidade.

**Experimento por Colocação 2:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados próximo e acima da suspensão, mais a velocidade.

**Experimento por Colocação 3:** Foram utilizados os dados de força de aceleração 3D e taxa de rotação 3D amostrados no painel de controle, mais a velocidade.

Para avaliar a capacidade de generalização do aprendizado de cada técnica, avaliando sua adaptabilidade para contextos desconhecidos onde há variação nas propriedades de dependência veicular, de condução e ambiental, os dados de treinamento e validação foram divididos

da mesma maneira que nos estudos das seções anteriores, separando-os de acordo com o conjunto de dados em três *experimentos por contexto*:

**Experimento por Contexto 1:** O modelo aprende dados de todos os veículos e motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (65%):** PVS 1, PVS 3, PVS 4, PVS 6, PVS 7, PVS 9.
- **Validação (35%):** PVS 2, PVS 5, PVS 8.

**Experimento por Contexto 2:** O modelo aprende dados de todos os cenários para alguns veículos e alguns motoristas; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 3, PVS 7, PVS 8, PVS 9.
- **Validação (34%):** PVS 4, PVS 5, PVS 6.

**Experimento por Contexto 3:** O modelo aprende dados de alguns veículos com alguns motoristas para alguns cenários; mas não todos os veículos com todos os motoristas para todos os cenários.

- **Treinamento (66%):** PVS 1, PVS 2, PVS 4, PVS 6, PVS 8, PVS 9.
- **Validação (34%):** PVS 3, PVS 5, PVS 7.

Neste estudo, para bem avaliar as variações contextuais, além de dados de lombadas em diferentes pavimentos, sendo asfalto e a paralelepípedo, os modelos ainda contam com amostras de vias de terra, as quais são importantes para os modelos minimizarem FP, uma vez que estradas de terra apresentam irregularidades que produzem assinaturas nos sinais semelhantes às produzidas por lombadas, sendo importante que os modelos saibam diferenciá-las.

Finalmente, para analisar a influência do número de amostras na classificação, foram criados três *experimentos por tamanho de janela de dados*, com janelas de 100, 200, 300, 400, e 500 amostras. A janela de dados utilizada foi deslizante com sobreposição total, e cada amostra correspondeu a um vetor com os valores das características. A aplicação de janela deslizante se mostra importante neste estudo para analisarmos o comportamento do modelo com entrada de dados em diferentes segmentos da lombada, como janelas com amostras do início, meio ou fim do obstáculo. Em resumo, cada experimento realizado neste estudo é um elemento do produto cartesiano entre *experimentos por colocação*, *experimentos por contexto* e *experimentos por tamanho de janela de dados*, conforme ilustra a Figura 41.

Através da Tabela 13 (pág. 83) observamos certo desbalanceamento das classes de dados. Sendo assim, se faz necessário analisar o grau desse desbalanceamento, uma vez que classes sub-representadas (*minor data class*) podem não ser adequadamente tratadas pelos modelos, resultando em viés. Para avaliar a necessidade de aplicar técnicas de balanceamento,



medimos a proporção de distribuição de cada classe de dados conforme detalhado Tabela 41. A distribuição de classes foi calculada sobre os dados de treinamento, uma vez que são com estes que os modelos aprendem (HE; MA, 2013; KUHN; JOHNSON et al., 2013). A distribuição foi calculada em relação ao tamanho da janela de dados. Sendo assim, o valor detalhado em cada célula corresponde ao limite inferior e superior da distribuição da classe de dados, uma vez que cada tamanho da janela de dados apresenta uma pequena variação deste valor.

Tabela 41 – Distribuição de classes de lombadas

Fonte de Dados	Classe de Dados			
	CL - Com Lombada		SL - Sem Lombada	
	Percentual	Proporção	Percentual	Proporção
Exp. por Contexto 1	1,61% - 1,45%	1:60,87 - 1:67,76	98,39% - 98,55%	1:0,01 - 1:0,02
Exp. por Contexto 2	1,58% - 1,36%	1:62,03 - 1:72,22	98,42% - 98,64%	1:0,01 - 1:0,02
Exp. por Contexto 3	1,63% - 1,44%	1:60,17 - 1:68,39	98,37% - 98,56%	1:0,01 - 1:0,02

Fonte: Desenvolvido pelo autor.

O desbalanceamento de classes pode ser considerado leve ou severo, onde as proporções de distribuição que variam de 1:4 até 1:100 (presença de 20% - 1%) são consideradas desbalanceamento leve e proporções de distribuição que variam de 1:100 ou mais (<1% de presença) são consideradas desbalanceamentos severos (KRAWCZYK, 2016; BROWNLEE, 2020). Analisando a Tabela 41, observamos que ambas as classes de dados apresentam desbalanceamentos leve, mas com alta tendência a desbalanceamentos severo. Para a classe *Com Lombada (CL)*, a proporção varia entre 1:60,17 a 1:72,22, onde para 1 amostra da classe de dados *CL*, há de 60,17 a 72,22 amostras da classe de dados *Sem Lombada (SL)*. Portanto, realizamos *downsampling* nos segmentos de dados sem lombada (*SL*), descartando as janelas de sobreposição. Em suma, os dados onde há lombada contam com janela deslizante com sobreposição total, e onde não há lombada, a janela é fixa sem sobreposição. Com esta abordagem, a distribuição obtida é detalhada na Tabela 42, com uma redução da desproporção em aproximadamente 10 vezes.

Tabela 42 – Distribuição de classes de lombadas após *downsampling*

Fonte de Dados	Classe de Dados			
	CL - Com Lombada		SL - Sem Lombada	
	Percentual	Proporção	Percentual	Proporção
Exp. por Contexto 1	88,01% - 62,12%	1:0,14 - 1:0,61	37,87% - 11,99%	1:1,64 - 1:7,34
Exp. por Contexto 2	87,34% - 61,68%	1:0,14 - 1:0,62	38,31% - 12,66%	1:1,61 - 1:6,90
Exp. por Contexto 3	87,86% - 62,41%	1:0,14 - 1:0,60	37,59% - 12,14%	1:1,66 - 1:7,28

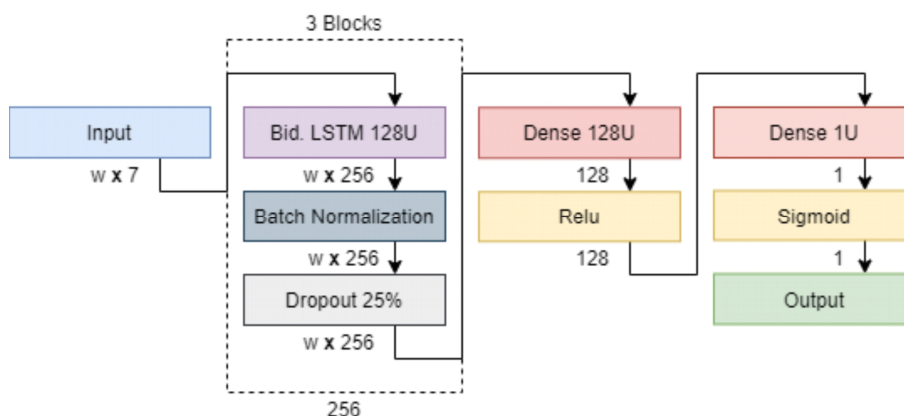
Fonte: Desenvolvido pelo autor.

## 9.2 PROCESSAMENTO

Após o pré-processamento, os dados foram aplicados em cinco modelos de *Deep Learning*, sendo baseados em LSTM, GRU, CNN, CNN-LSTM e ConvLSTM. Todos os modelos desenvolvidos são sequenciais e utilizam o otimizador Adam em conjunto com a função de perda Entropia Cruzada Binária. Todos os modelos tiveram seus hiperparâmetros ajustados (*hyperparameter tuning*) utilizando o algoritmo *HyperBand* implementado na biblioteca Keras Tuner, onde foram testados diferentes tipos de camadas, número de camadas e neurônios, funções de ativação e várias outras características para encontrar o conjunto ideal de hiperparâmetros para cada DNN.

O melhor modelo baseado em LSTM obtido através do ajuste de hiperparâmetros é detalhado na Figura 48. O modelo DNN é composto de um bloco de camada de entrada, três blocos de camadas de recorrência e regularização e dois blocos de camada totalmente conectados para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, onde *janelas* são os agrupamentos de todas as janelas de dados, *sequências* são as sequências de dados para cada janela, e *características* são os valores das variáveis de entrada. Cada bloco de recorrência e regularização é composto por uma camada LSTM bidirecional de 128 unidades, seguida por uma camada de *Batch Normalization* e uma camada de *Dropout* em 25%. Após o processamento nas camadas recorrentes, os parâmetros são passados para os blocos totalmente conectados, onde existem duas camadas *Dense*, a primeira com 128 neurônios e ativação *Relu*, e a segunda com 1 neurônio e ativação *Sigmoid*. A saída produzida é uma resposta binária à presença ou ausência de lombadas na janela de dados. A saída esperada são os rótulos mais presentes na janela de dados.

Figura 48 – Modelo LSTM para detecção de lombadas

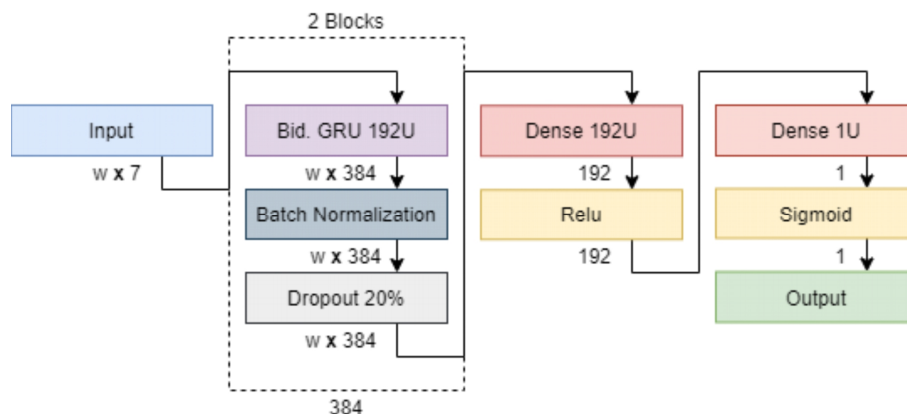


Fonte: Desenvolvido pelo autor.

O melhor modelo baseado em GRU obtido neste estudo está detalhado na Figura 49. O modelo de DNN é composto de um bloco de camada de entrada, dois blocos de camadas de recorrência e regularização e dois blocos de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x*

*características*, semelhante ao baseado em LSTM. Cada bloco de recorrência e regularização é composto por uma camada GRU bidirecional de 192 unidades, seguida por uma camada de *Batch Normalization* e uma camada de *Dropout* em 20%. Após o processamento nas camadas recorrentes, os parâmetros passam para um bloco totalmente conectado, onde existem duas camadas *Dense*, a primeira com 192 neurônios e ativação *Relu*, e a segunda com 1 neurônio e ativação *Sigmoid*, produzindo a resposta binária.

Figura 49 – Modelo GRU para detecção de lombadas

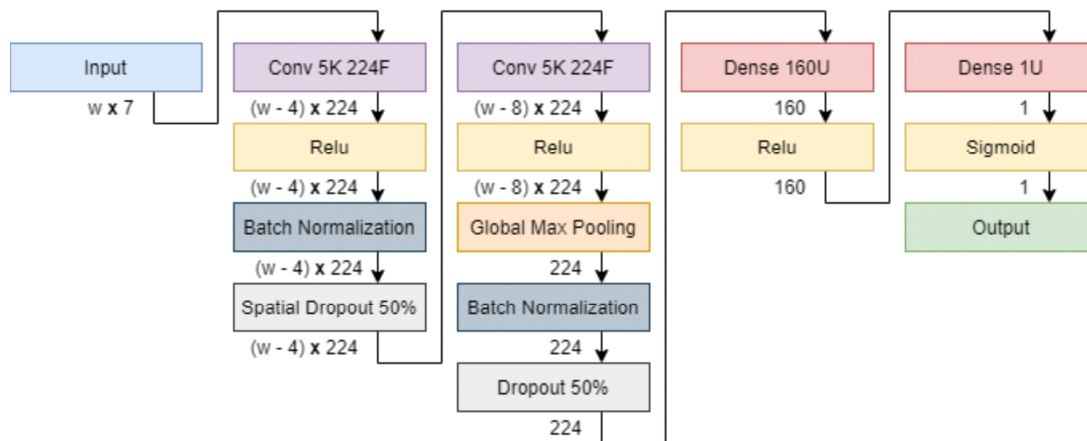


Fonte: Desenvolvido pelo autor.

O melhor modelo baseado em CNN obtido é detalhado na Figura 50. O modelo de DNN é composto de um bloco de entrada, dois blocos de convolução e regularização e dois blocos de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x características*, semelhante aos modelos baseados em LSTM e GRU. O primeiro bloco de convolução e regularização é composto por uma camada Conv 1D com 224 filtros com *kernel* de tamanho 5 para extração de características e ativação *Relu*, seguida de regularização através de uma camada de *Batch Normalization* e uma camada de *Spatial Dropout 1D* em 50%. O último bloco de convolução e regularização tem uma camada Conv 1D com as mesmas configurações da anterior, uma camada *Global Max Pooling 1D* para extrair recursos mais robustos por meio dos valores máximos em cada região, e camadas de regularização por *Batch Normalization* e *Dropout* em 50%. Finalmente, os blocos totalmente conectados consistem em duas camadas *Dense*, uma com 160 neurônios e ativação *Relu* e outra com 1 neurônio e ativação *Sigmoid*, produzindo a saída binária.

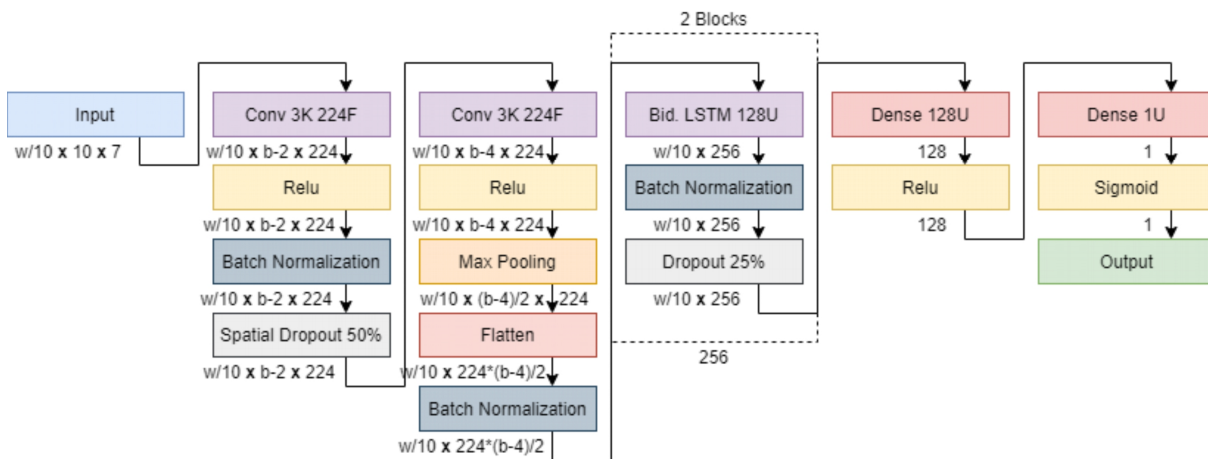
O melhor modelo híbrido baseado em CNN-LSTM obtido é detalhado na Figura 51. O modelo de DNN baseado em CNN-LSTM desenvolvido é composto de um bloco de entrada, dois blocos de convolução e regularização, dois blocos recorrentes e de regularização e dois blocos de camada totalmente conectadas para produção de saída. O bloco de entrada tem uma camada que recebe um tensor *janelas x sequências x subsequências x características*, onde *janelas* são os agrupamentos de todas as janelas de dados, *sequências* são as sequências de dados para cada janela, *subsequências* são as subpartes da sequência de dados original, e *características* os valores das variáveis de entrada. O primeiro bloco de convolução e regularização é

Figura 50 – Modelo CNN para detecção de lombadas



Fonte: Desenvolvido pelo autor.

Figura 51 – Modelo CNN-LSTM para detecção de lombadas



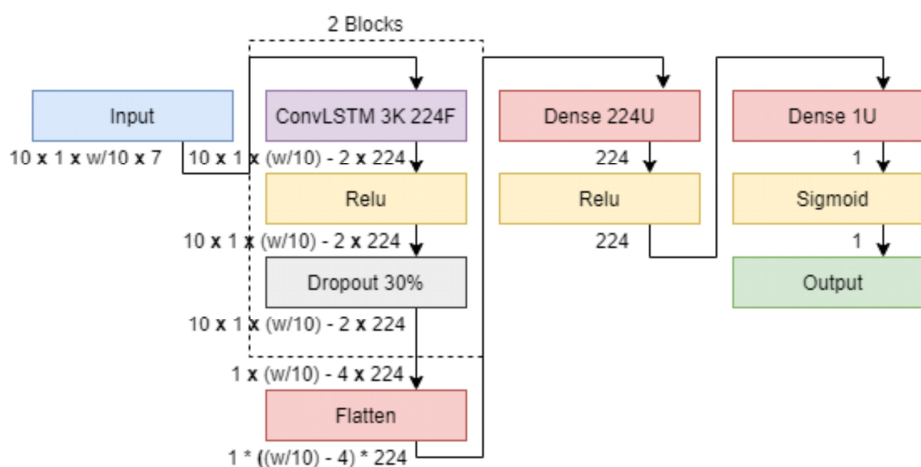
Fonte: Desenvolvido pelo autor.

composto por uma camada Conv 1D com 224 filtros com *kernel* de tamanho 3 para extração de características e ativação *Relu*, seguida de regularização através de uma camada de *Batch Normalization* e uma camada de *Spatial Dropout 1D* em 50%. O último bloco de convolução e regularização tem uma camada Conv 1D com as mesmas configurações da anterior, uma camada *Max Pooling 1D* para extrair recursos mais robustos por meio dos valores máximos em cada região, uma camada *Flatten* para reagrupar os recursos extraídos na sequência temporal original, e regularização por camada de *Batch Normalization*. Em seguida, cada bloco de recorrência e regularização é composto por uma camada LSTM bidirecional de 128 unidades, seguida por uma camada de *Batch Normalization* e uma camada de *Dropout* em 25%. Finalmente, os parâmetros resultantes passam para duas camadas *Dense*, a primeira com 128 neurônios e ativação de *Relu*, e a segunda com 1 neurônio e ativação de *Sigmoid*, produzindo a saída binária.

O melhor modelo baseado em ConvLSTM obtido é detalhado na Figura Figura 52. O modelo DNN é composto de um bloco de entrada, dois blocos convolucionais recorrentes e de

regularização, e dois blocos de camadas totalmente conectadas para produção de saída. O bloco de entrada possui uma camada que recebe um tensor *janelas x sequências x subsequências x características*, semelhante ao modelo CNN-LSTM. Os dois blocos convolucionais recorrentes e de regularização são compostos por uma camada *ConvLSTM 1D* com 224 filtros *kernel* de tamanho 3 e ativação *Relu*, seguido por regularização através de uma camada *Dropout* em 30%. Finalmente, os parâmetros resultantes são achatados (*flattened*) e passados para duas camadas *Dense*, a primeira com 224 neurônios e ativação *Relu*, e a segunda com 1 neurônio e ativação *Sigmoid*, produzindo a saída binária.

Figura 52 – Modelo ConvLSTM para detecção de lombadas



Fonte: Desenvolvido pelo autor.

Os demais hiperparâmetros das técnicas de *Deep Learning* foram utilizados com seus valores padrões, disponíveis em TensorFlow (2021d), TensorFlow (2021c), TensorFlow (2021a) e TensorFlow (2021b).

### 9.3 ANÁLISE DE RESULTADOS

Neste estudo todos os modelos de detecção de lombadas foram desenvolvidos na linguagem Python 3, utilizando da biblioteca Keras 2, a qual é uma API de alto nível do TensorFlow. Os hiperparâmetros dos modelos foram afinados através da biblioteca Keras Tuner. Todos os experimentos foram executados em máquinas do Google *Collaboratory*, de mesma configuração dos estudos das seções anteriores. Cada experimento é um elemento do produto Cartesiano entre *experimentos por colocação*, *experimentos por contexto* e *experimentos por tamanho da janela de dados*, sendo executado três vezes e recuperando-se o melhor das três execuções. Consideramos a melhor execução aquela com maior valor de acurácia na fase de validação, uma vez que o treinamento dos modelos foi configurado para maximizar a métrica de acurácia.

Os resultados obtidos com a execução de todos os experimentos são apresentados nas Tabelas 43 a 47. Em cada tabela são detalhados os resultados para determinado modelo de

DNN, para cada colocação, contexto e janela de dados. Também são apresentadas métricas de média aritmética para cada tipo de experimento, sendo Acurácia Média dos Experimentos por Contexto (AMEC) e Acurácia Média dos Experimentos por Contexto e Colocação (AMECC), onde são destacados os menores e maiores valores de acurácia obtidos na fase de validação.

Tabela 43 – Valores de acurácia em validação obtidos pelo modelo LSTM

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	95,03%	96,53%	98,59%	99,18%	99,21%
	2	90,70%	94,33%	97,08%	97,12%	97,81%
	3	95,05%	97,54%	98,66%	99,12%	98,52%
	AMEC	93,59%	96,13%	98,11%	98,47%	98,52%
Próximo e Acima da Suspensão	1	92,77%	93,68%	97,83%	98,97%	98,46%
	2	92,90%	94,98%	97,21%	97,76%	97,66%
	3	95,00%	96,64%	98,60%	99,25%	99,30%
	AMEC	93,56%	95,10%	97,88%	98,66%	98,47%
Painel de Controle	1	93,99%	94,55%	99,22%	98,77%	98,18%
	2	92,10%	95,70%	97,37%	97,38%	98,48%
	3	95,19%	97,36%	99,24%	99,33%	98,62%
	AMEC	93,76%	95,87%	98,61%	98,49%	98,43%
	AMECC	93,64%	95,70%	98,20%	98,54%	98,47%

Fonte: Desenvolvido pelo autor.

Tabela 44 – Valores de acurácia em validação obtidos pelo modelo GRU

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	94,23%	96,42%	97,84%	98,60%	95,78%
	2	92,76%	94,20%	96,83%	96,54%	97,25%
	3	94,46%	97,54%	98,55%	98,88%	96,17%
	AMEC	93,82%	96,05%	97,74%	98,01%	96,40%
Próximo e Acima da Suspensão	1	93,57%	93,61%	98,23%	94,44%	95,45%
	2	90,93%	94,17%	97,17%	97,39%	96,45%
	3	93,90%	96,82%	98,53%	98,00%	98,62%
	AMEC	92,80%	94,87%	97,98%	96,61%	96,84%
Painel de Controle	1	94,41%	96,78%	97,94%	96,88%	95,92%
	2	92,05%	96,37%	96,91%	98,05%	93,01%
	3	94,64%	96,93%	98,54%	99,18%	97,38%
	AMEC	93,70%	96,69%	97,80%	98,04%	95,44%
	AMECC	93,44%	95,87%	97,84%	97,55%	96,23%

Fonte: Desenvolvido pelo autor.

Tabela 45 – Valores de acurácia em validação obtidos pelo modelo CNN

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	94,20%	95,12%	97,38%	97,45%	99,8%
	2	91,02%	90,57%	93,32%	95,33%	95,73%
	3	95,55%	96,07%	97,62%	98,32%	99,36%
	AMEC	93,59%	93,92%	96,11%	97,03%	98,30%
Próximo e Acima da Suspensão	1	95,1%	96,08%	97,18%	97,55%	98,85%
	2	93,77%	94,40%	96,96%	97,66%	97,11%
	3	95,11%	95,94%	98,67%	99,40%	97,84%
	AMEC	94,66%	95,48%	97,60%	98,20%	97,93%
Painel de Controle	1	95,41%	95,69%	97,61%	97,84%	99,78%
	2	94,43%	93,59%	96,24%	97,19%	97,41%
	3	95,57%	95,11%	97,51%	98,52%	98,96%
	AMEC	95,14%	94,80%	97,12%	97,85%	98,72%
	AMECC	94,46%	94,73%	96,94%	97,69%	98,32%

Fonte: Desenvolvido pelo autor.

Tabela 46 – Valores de acurácia em validação obtidos pelo modelo CNN-LSTM

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	94,49%	96,76%	98,95%	99,08%	98,20%
	2	94,30%	94,63%	96,83%	97,72%	96,79%
	3	95,13%	97,75%	98,89%	98,92%	98,93%
	AMEC	94,64%	96,38%	98,22%	98,57%	97,97%
Próximo e Acima da Suspensão	1	94,14%	96,63%	98,75%	99,25%	99,79%
	2	92,78%	94,99%	97,33%	97,39%	97,67%
	3	94,92%	96,77%	99,21%	99,11%	99,38%
	AMEC	93,95%	96,13%	98,43%	98,59%	98,94%
Painel de Controle	1	94,18%	96,46%	99,1%	99,22%	99,73%
	2	92,74%	96,2%	97,37%	97,44%	97,46%
	3	94,86%	96,89%	99,13%	98,76%	99,41%
	AMEC	93,93%	96,52%	98,53%	98,48%	98,87%
	AMECC	94,17%	96,34%	98,39%	98,55%	98,59%

Fonte: Desenvolvido pelo autor.

Tabela 47 – Valores de acurácia em validação obtidos pelo modelo ConvLSTM

Tipo de Experimento		Tamanho da Janela de Dados				
Colocação	Contexto	100	200	300	400	500
Próximo e Abaixo da Suspensão	1	93,77%	96,21%	97,91%	98,43%	98,30%
	2	91,57%	94,92%	97,19%	96,22%	95,93%
	3	94,82%	96,73%	98,57%	98,74%	97,12%
	AMEC	93,39%	95,95%	97,89%	97,79%	97,12%
Próximo e Acima da Suspensão	1	92,92%	96,12%	97,86%	98,21%	98,95%
	2	92,72%	96,60%	97,65%	96,93%	97,52%
	3	94,24%	96,39%	98,48%	99,30%	99,17%
	AMEC	93,29%	96,37%	98,00%	98,15%	98,55%
Painel de Controle	1	94,95%	96,68%	98,41%	99,28%	98,79%
	2	93,25%	95,90%	97,31%	98,92%	98,36%
	3	95,04%	97,36%	98,77%	99,35%	99,66%
	AMEC	94,41%	96,65%	98,16%	99,18%	98,94%
	AMECC	93,70%	96,32%	98,02%	98,37%	98,20%

Fonte: Desenvolvido pelo autor.

Em nossa análise, para avaliar a habilidade dos modelos generalizarem seu aprendizado para contextos desconhecidos, como diferentes veículos, motoristas ou ambientes, consideramos que o modelo deve obter bom desempenho nos três *experimentos por contexto*. Sendo assim, nossa análise é pautada inicialmente na média de acurácia destes experimentos, representada pela métrica AMEC. Através dos valores AMEC detalhados nas Tabelas 43-47, observamos que todos os modelos de DNN desenvolvidos obtiveram bons resultados independentes do contexto, com o valor de média de acurácia variando entre 92,80% e 99,18% conforme modelo e ponto de coleta de dados.

Analisando o impacto do tamanho das janelas de dados, observamos que as janelas de 100 e 200 amostras obtiveram todos os piores resultados para todos os experimentos e modelos, denotando ser uma quantidade de amostras insuficientes para reconhecer lombadas com uma boa confiabilidade. Nestas duas janelas estão as maiores variações dos valores de acurácia, onde a janela de dados de 100 amostras obteve acurácia média de 93,70%, e a de 200 amostras obteve 95,87%. Por outro lado, as janelas de 300, 400 e 500 amostras obtiveram os melhores resultados, com uma estabilização dos valores de acurácia, apresentando uma variação muito pequena. Na média entre todos os experimentos e modelos, a janela de 300 amostras obteve acurácia de 98,02%, a de 400 obteve 98,37%, e a de 500 obteve 98,32%. Neste estudo, uma lombada inteira possui entre 223 a 419 amostras. Logo, podemos concluir que os modelos necessitam de que a janela de dados contenha a lombada por inteiro o muito próximo disso para obter melhores resultados.

Em relação aos pontos de coleta de dados, todos obtiveram bons resultados, com pequena variação de uma colocação para outra. Os valores de média acurácia AMEC entre os modelos para a colocação próximo e abaixo da suspensão variaram de 93,39% até 98,57%;



para próximo e acima da suspensão de 92,80% até 98,94%; e no painel de controle de 93,70% até 99,18%. A escolha do melhor modelo neste ponto pode levar em consideração a aplicação final, onde o modelo baseado em CNN-LSTM se mostra melhor para quando os sensores são empregados próximo e abaixo da suspensão ou próximo e acima da suspensão, enquanto que o modelo baseado em ConvLSTM se mostra melhor para aplicação no painel de controle. Contudo, neste estudo consideramos que o melhor modelo deve possibilitar sua operação independentemente do local de colocação dos sensores no veículo. Portanto, deve ser aquele com melhor desempenho entre os diferentes pontos de coleta, representado pela média de acurácia entre os *experimentos por colocação*. Sendo assim, esta métrica é detalhada nas tabelas por AMECC, onde para determinar a melhor configuração de janela para cada modelo é considerado a média de acurácia entre todos os *experimentos por colocação* e *experimentos por contexto*.

Através da métrica supracitada, observamos que o melhor modelo é o baseado em CNN-LSTM em janela de 500 amostras, resultando em média de acurácia de 98,59%. O segundo melhor é o baseado em LSTM com 400 amostras resultando em 98,54%. O terceiro o baseado em ConvLSTM de 400 amostras com 98,37%. Em quarto o baseado em CNN de 500 amostras com 98,32%. Por último, o modelo baseado em GRU de 300 amostras resultando em 97,84%. A melhor configuração de cada modelo é detalhada na Tabela 48 com as demais métricas de avaliação. Todos os valores das métricas apresentadas nesta tabela correspondem a média entre os *experimentos por colocação* e *experimentos por contexto*.

Tabela 48 – Métricas de avaliação para a melhor configuração de cada modelo DNN

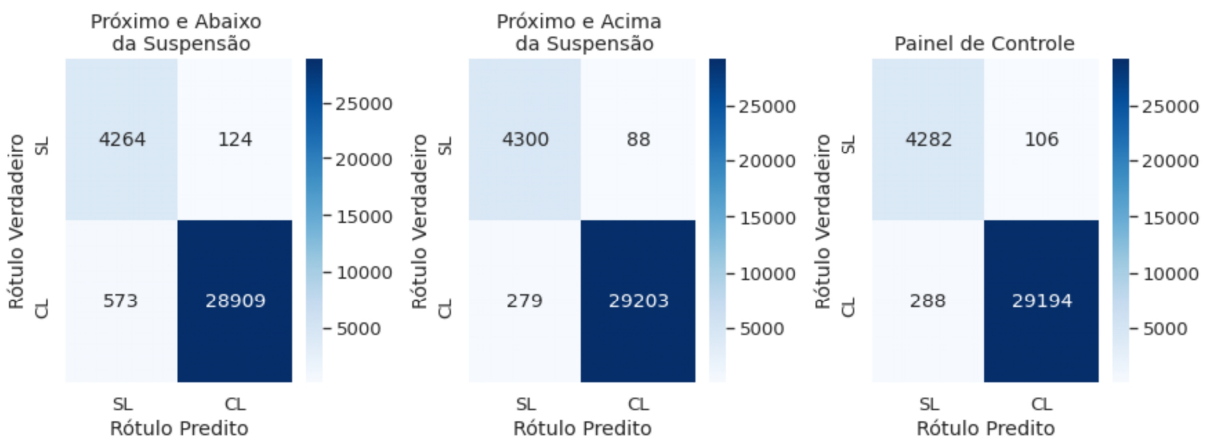
Métrica de Avaliação	Modelo	LSTM	GRU	CNN	CNN-LSTM	ConvLSTM
	Janela	400	300	500	500	400
Acurácia	Treinamento	99,21%	99,71%	99,11%	99,53%	99,56%
	Validação	98,54%	97,84%	98,32%	98,59%	98,37%
Precisão	CL	99,25%	99,64%	99,67%	99,64%	99,22%
	SL	94,40%	90,40%	90,57%	92,34%	93,49%
	Média	96,83%	95,02%	95,12%	95,99%	96,36%
Recall	CL	99,06%	97,73%	98,40%	98,75%	98,89%
	SL	95,32%	98,36%	97,83%	97,59%	95,13%
	Média	97,19%	98,05%	98,12%	98,17%	97,01%
F1-Score	CL	99,15%	98,67%	99,02%	99,19%	99,05%
	SL	94,77%	94,18%	93,83%	94,74%	94,18%
	Média	96,96%	96,43%	96,43%	96,97%	96,62%

Fonte: Desenvolvido pelo autor.

Na Tabela 48 são apresentadas as métricas de acurácia, precisão, *recall* e *f1-score* para cada melhor configuração dos modelos. Embora a acurácia seja amplamente utilizada, em conjuntos de dados com desbalanceamento ela pode esconder viés, especialmente quando a classe minoritária não é bem tratada pelo modelo (HE; MA, 2013). Para isto são utilizadas as

métricas de avaliação precisão, *recall* e *f1-score*, analisando a performance do modelo em cada classe de dados. Na tabela, para cada métrica são apresentadas os valores para as duas classes de dados deste estudo: *Com Lombada* (CL) e *Sem Lombada* (SL). Como podemos observar, todas as classes de dados foram corretamente tratadas por todos os modelos sem haver viés, de forma que a técnica de *downsampling* empregada foi efetiva. Todas as métricas obtiveram valores acima de 90%. Neste estudo, consideramos que o melhor modelo deve ser aquele que maximize tanto os VP quanto os VN, e minimize os FN e FP. Desta forma, o modelo deve ter a melhor performance tanto na detecção correta de lombadas quanto a detecção correta de não lombadas. Sendo assim, a métrica de *f1-score* associa estes dois fatores, e na média para as duas classes de dados o modelo baseado em CNN-LSTM com janela de 500 amostras foi o que obteve melhor resultado. Portanto, consideramos o modelo baseado em CNN-LSTM o melhor modelo para detecção de lombadas, com valor de média entre *experimentos por colocação* e *experimentos por contexto* de 98,59% para acurácia, 95,99% para precisão, 98,17% para *recall* e 96,97% para *f1-score*. Na Figura 53 é ilustrado a matriz confusão para o modelo baseado em CNN-LSTM em cada ponto de coleta de dados.

Figura 53 – Matriz de confusão para o modelo CNN-LSTM em cada ponto de coleta de dados



Fonte: Desenvolvido pelo autor.

## 10 MATERIAIS RESULTANTES

A realização deste trabalho resultou em diversos materiais disponíveis publicamente na página do projeto no Github <sup>1 2</sup>. Os seguintes materiais de *software* foram produzidos:

**Módulo de Coleta de Dados e Pré-Processamento:** contém o código-fonte dos projetos utilizados para coleta e ajuste dos dados, tal como gerenciamento de sensores para amostragem, armazenamento dos sinais, sincronização de sensores multimodais, fusão de dados, criação de GT, visualização de dados brutos, criação de vídeos integrando dados de MPU, GPS e câmera, etc.

**Módulo de Conjuntos de Dados:** contém todos os conjuntos de dados PVS produzidos.

**Módulo de Classificação de Tipo de Superfície de Pista 1:** código-fonte com todos os modelos desenvolvidos para o primeiro estudo de classificação de tipo de superfície, juntamente com todos os experimentos executados, permitindo pesquisas futuras executarem, compararem e auditarem.

**Módulo de Classificação de Tipo de Superfície de Pista 2:** código-fonte com todos os modelos desenvolvidos para o segundo estudo de classificação de tipo de superfície, juntamente com todos os experimentos executados, permitindo pesquisas futuras executarem, compararem e auditarem.

**Módulo de Classificação de Qualidade de Superfície de Pista:** código-fonte com todos os modelos desenvolvidos para o estudo de classificação de qualidade de superfície, juntamente com todos os experimentos executados, permitindo pesquisas futuras executarem, compararem e auditarem.

**Módulo de Reconhecimento de Lombadas:** código-fonte com todos os modelos desenvolvidos para o estudo de detecção de lombadas, juntamente com todos os experimentos executados, permitindo pesquisas futuras executarem, compararem e auditarem.

**Módulo de Melhores Modelos de Percepção Veicular:** código-fonte simplificado para uso dos melhores modelos desenvolvidos, sendo eles:

- Classificação de Tipo de Superfície de Pista: modelo CNN, classificando em terra, paralelepípedo e asfalto.
- Classificação de Qualidade de Superfície de Pista: modelo CNN, classificando em bom, regular e ruim.
- Reconhecimento de Lombadas: modelo CNN-LSTM, detectando lombadas.

Todos os módulos descritos compõem o programa *Intelligent Road Assessment System* (IRAS), o qual já possui Registro de Programa de Computador<sup>3</sup> (INPI) no Instituto Nacional da Propriedade Industrial, sendo distribuído sob a licença **CC BY-NC-ND 4.0**. Em relação ao *hardware* e rede de sensores desenvolvida, pedido de patente está em análise. Para promover a pesquisa e auxiliar na popularização deste tipo de sensoriamento em ITS, foi criado um canal no Youtube <sup>4</sup>, com divulgação de apresentações, vídeos didáticos e vídeos com os melhores

<sup>1</sup> <https://github.com/Intelligent-Vehicle-Perception/Intelligent-Vehicle-Perception-Based-on-Inertial-Sensing-and-Artificial-Intelligence>

<sup>2</sup> <https://codigos.ufsc.br/lapix/intelligent-vehicle-perception-based-on-inertial-sensing>

<sup>3</sup> Processo número BR512021000667-4

<sup>4</sup> <https://www.youtube.com/channel/UCoWkXjgUNhGCYieR-ys-IBw>

modelos em execução produzindo seus reconhecimentos e classificações. Também estão em produção artigos acessíveis para público não especializado, que serão publicados no *Medium* e *Towards Data Science*. Em relação às publicações científicas, foram produzidos os seguintes artigos:

***Vehicular Perception and Proprioception Based on Inertial Sensing: a Systematic Review***

**Editora:** Brazilian National Institute for Digital Convergence.

**Status:** Publicado.

**Ano:** 2018.

**Qualis:** C.

**Classificação de Qualidade de Superfície de Pista Baseado em Sensoriamento Inercial e Lógica**

*Fuzzy*

**Editora:** UNIVALI. Anais do *Computer on the Beach*.

**Status:** Publicado.

**Ano:** 2019.

**Qualis:** B4.

***Vehicular Perception Based on Inertial Sensing - A Structured Mapping of Approaches and Methods***

**Editora:** Springer. Revista *SN Computer Science*.

**Status:** Publicado.

**Ano:** 2020.

**Qualis:** Indefinido.

***Multi-Contextual and Multi-Aspect Analysis for Road Surface Type Classification Through Inertial Sensors and Deep Learning***

**Editora:** IEEE Xplore. Anais do *2020 X Brazilian Symposium on Computing Systems Engineering (SBESC)*.

**Status:** Publicado.

**Ano:** 2020.

**Qualis:** B2.

***Road Surface Type Classification Based on Inertial Sensors and Machine Learning: A Comparison Between Classical and Deep Machine Learning Approaches For Multi-Contextual Real-world Scenarios***

**Editora:** Springer. Revista *Computing*.

**Status:** Publicado.

**Ano:** 2021.

**Qualis:** B1.

***Speed Bump Detection Through Inertial Sensors and Deep Learning in a Multi-Contextual Analysis***

**Editora:** Springer. Revista *Neural Processing Letters*.

**Status:** Com o Editor.

**Ano:** 2021.

**Qualis:** A2.

## 11 CONCLUSÕES E DISCUSSÕES

Os sistemas de transporte se estabeleceram ao longo da história como importante condicionante do desenvolvimento humano. Devido a sua vital importância surgiram os ITS para tornar as redes de transporte mais eficientes e seguras. As aplicações em ITS já estão presentes no cotidiano, tais como ADAS, e tendem a ser cada vez mais comuns em breve, com veículos autônomos. Contudo, para que estas aplicações operem de forma confiável, mostra-se necessário o emprego de tecnologias seguras na coleta de dados brutos; a diversificação e redundância destas fontes de dados; o desenvolvimento de modelos de IA que operem corretamente em contextos reais; e da geração de uma grande variedade de informações situacionais sobre o ambiente de trânsito e seus participantes.

As diversas tecnologias para coleta de dados brutos empregados na geração de informações situacionais do modal de transporte terrestre podem ser classificadas entre abordagem passiva e ativa. Dentre as de abordagem passiva, os sensores inerciais se mostram uma tecnologia segura, não poluente, de baixo custo e de fácil instalação, sendo ideais para uso em larga escala em ITS. Com estes sensores, é possível produzir uma grande variedade de informações na forma de percepções veiculares, dentre exterocepções e propriocepções, tais como as três trabalhadas nesta pesquisa: tipo de superfície de pista, qualidade de superfície de pista e detecção de lombadas. Contudo, ao contrário de outras tecnologias, tais como câmera ou LIDAR, as soluções baseadas em sensores inerciais levantadas através da nossa RSL mostram-se sumariamente provas de conceito simplistas, que não consideram variações das condições contextuais nas quais a solução será submetida quando aplicada em cenários do mundo real. Desta forma, mostram-se inaptas a se adaptar a diferentes contextos e, portanto, não são confiáveis para sua ampla utilização.

Baseado no problema supracitado, consideramos que a adaptabilidade é um fator essencial para prover a confiabilidade necessária que possibilite uma ampla aplicação de soluções com sensores inerciais em ITS. Sendo assim, neste estudo produzimos uma metodologia para desenvolvimento e validação de modelos de percepção veicular que operem de forma adaptativa, ou seja, que consigam generalizar seu aprendizado para contextos desconhecidos. Com este objetivo, identificamos inicialmente os fatores de dependência que afetam os dados dos sensores inerciais e, conseqüentemente, o processo de aprendizado e generalização dos modelos. Esses fatores foram agrupados em quatro propriedades, sendo elas sensoriais, veiculares, ambientais e de condução. Por meio do estabelecimento das propriedades de dependência, realizamos várias coletas de dados para obter variações desses fatores. As variações incluíram dados coletados em três diferentes colocações no veículo, em três veículos diferentes, dirigidos por três motoristas, trafegando em três tipos de superfície de pista, as quais apresentaram variações no estado de conservação, além da presença de obstáculos e anomalias.

Com os dados coletados, produzimos um *design* experimental para validar nossa hipótese inicial, a qual considera que através de conjuntos de dados que bem representem a diversidade contextual envolvida na aplicação dos sensores inerciais em ITS para geração de percepção

veicular, ou seja, que os dados contenham variações contextuais significativas em relação aos fatores de dependência da solução, é possível construir modelos de IA capazes de aprender as relações e a influências dos fatores de dependência nos sinais dos sensores, possibilitando a generalização de seu aprendizado para cenários desconhecidos de forma confiável. Além deste aspecto, nossos experimentos analisaram também a utilização de diferentes domínios de análise, características de entrada, tamanho da janela de dados, e o impacto dado o ponto de coleta no veículo. Adicionalmente, analisamos a aplicação de técnicas com diferentes abordagens, dentre técnicas de Aprendizado de Máquina clássico e *Deep Learning*, para validar a mais apropriada.

Em uma análise macro, as técnicas baseadas em *Deep Learning* se mostraram consideravelmente superiores quando comparadas as técnicas clássicas, evidenciando a capacidade desta abordagem de extrair características de alto nível diretamente dos dados brutos que bem representassem o problema tratado, e seu excelente aprendizado em camadas com base nesses parâmetros. Todos os melhores modelos resultantes são baseados em *Deep Learning*. Para classificação de tipo de superfície entre terra, paralelepípedo e asfalto, a rede CNN modelada classificou os segmentos com 93,04% de acurácia em validação para os dados de próximo e abaixo da suspensão; 92,02% para próximo e acima da suspensão; e 93,05% para o painel de controle. Para classificação de qualidade de superfície de pista entre boa, regular ou ruim, a rede CNN modelada classificou os segmentos com acurácia em validação de 93,62% para os dados de próximo e abaixo da suspensão; 93,90% para próximo e acima da suspensão; e de 93,04% para o painel de controle. Por fim, no reconhecimento de lombadas, a rede híbrida CNN-LSTM identificou o obstáculo com acurácia em validação de 97,97% para os dados de próximo e abaixo da suspensão; 98,94% para próximo e acima da suspensão; e 98,87% para o painel de controle. Os valores de acurácia correspondem a média obtida entre experimentos por variação de contexto, onde o modelo treinado foi submetido a um veículo, motorista ou ambiente desconhecido. Com base nos resultados, podemos observar que os modelos conseguiram apreender e generalizar corretamente as percepções independente do ponto de coleta no veículo ou do desconhecimento de características contextuais. Sendo assim, a hipótese deste estudo é validada, de forma que os modelos se mostraram confiáveis para generalizar seu aprendizado para contextos desconhecidos, diferentes daqueles nos quais o aprendizado ocorreu.

## 11.1 TRABALHOS FUTUROS

Com a disponibilização de todo o material deste estudo, como conjuntos de dados, códigos-fonte e modelos treinados, pesquisas futuras podem evoluir em diferentes frentes. A partir dos conjuntos de dados que produzimos, novos estudos podem ser realizados sobre a previsão ou classificação dos mais diversos padrões em ITS. Relacionado à exterocepção, podem ser reconhecidos buracos, rachaduras, trincas, relevo, entre outros. Relacionados à propriocepção, podem ser reconhecidos eventos de condução como aceleração, frenagem, virando à direita ou esquerda, assim como o nível de segurança de condução.

Em relação aos modelos de percepção veicular produzidos nesta pesquisa, novos estu-

dos podem analisar o comportamento do modelo com dados de diferentes pontos de coleta utilizados em conjunto; realizar experimentos com a permutação dos conjuntos de dados agrupados por fatores de dependência; desenvolver modelos com aprendizado contínuo; experimentar novas características de entrada no domínio da frequência ou tempo-frequência; realizar segmentação das amostras em função do tempo/distância utilizando *downsampling* quando existir mais amostras que o necessário para a entrada da DNN, ou *padding* quando a janela for menor que a necessária; utilizar nossos modelos de DNN como GT para treinar modelos baseados em visão computacional; etc. Novas coletas também podem ser produzidas para investigar outros tipos de superfície, como areia, lajotas sextavadas, paver, etc., assim como novas aplicações, tal como avaliação de piso de fábrica na Indústria 4.0.





## REFERÊNCIAS

- ABDELRAHEEM, M.; EL-MELEGY, M. Drive-by road condition assessment using internet of things technology. In: **2019 International Conference on Advances in the Emerging Computing Technologies (AECT)**. [S.l.: s.n.], 2020. p. 1–6.
- AFRIN, M.; MAHMUD, M. R.; RAZZAQUE, M. A. Real time detection of speed breakers and warning system for on-road drivers. In: **2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)**. [S.l.: s.n.], 2015. p. 495–498.
- AHMAD, W.; KAZMI, B. M.; ALI, H. Human activity recognition using multi-head cnn followed by lstm. In: **2019 15th International Conference on Emerging Technologies (ICET)**. [S.l.: s.n.], 2019. p. 1–6.
- AKHTAR, N.; PANDEY, K.; GUPTA, S. Mobile application for safe driving. In: **2014 Fourth International Conference on Communication Systems and Network Technologies**. [S.l.: s.n.], 2014. p. 212–216.
- AL-YAFEAI, D.; DARABSEH, T.; MOURAD, A. I. Quarter vs. half car model energy harvesting systems. In: **2019 Advances in Science and Engineering Technology International Conferences (ASET)**. [S.l.: s.n.], 2019. p. 1–5.
- ALAM, M. Y. et al. Crowdsourcing from the true crowd: Device, vehicle, road-surface and driving independent road profiling from smartphone sensors. **Pervasive and Mobile Computing**, v. 61, p. 101103, 2020. ISSN 1574-1192. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1574119219304687>.
- ALEMAYOH, T. T.; LEE, J. H.; OKAMOTO, S. Deep learning based real-time daily human activity recognition and its implementation in a smartphone. In: **2019 16th International Conference on Ubiquitous Robots (UR)**. [S.l.: s.n.], 2019. p. 179–182.
- ALJAAFREH, A. et al. Fuzzy inference system for speed bumps detection using smart phone accelerometer sensor. **Journal of Telecommunication, Electronic and Computer Engineering (JTEC)**, v. 9, n. 2-7, p. 133–136, 2017.
- ALLOUCH, A. et al. Roadsense: Smartphone application to estimate road conditions using accelerometer and gyroscope. **IEEE Sensors Journal**, v. 17, n. 13, p. 4231–4238, July 2017. ISSN 1530-437X.
- ALQUDAH, Y. A.; SABABHA, B. H. A statistical approach to estimating driving events by a smartphone. In: **2016 International Conference on Computational Science and Computational Intelligence (CSCI)**. [S.l.: s.n.], 2016. p. 1021–1025.
- ANDRIA, G. et al. Development of an automotive data acquisition platform for analysis of driving behavior. **Measurement**, v. 93, p. 278 – 287, 2016. ISSN 0263-2241. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0263224116303943>.
- ARAGÓN, M. E. et al. A machine learning pipeline to automatically identify and classify roadway surface disruptions. In: **Proceedings of the Sixteenth Mexican International Conference on Computer Science**. New York, NY, USA: Association for Computing Machinery, 2016. (ENC '16). ISBN 9781450348645. Disponível em: <https://doi.org/10.1145/3149235.3149238>.

ARROYO, C.; BERGASA, L. M.; ROMERA, E. Adaptive fuzzy classifier to detect driving events from the inertial sensors of a smartphone. In: **2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2016. p. 1896–1901. ISSN 2153-0017.

ASTARITA, V. et al. A mobile application for road surface quality control: Uniqualroad. **Procedia - Social and Behavioral Sciences**, v. 54, p. 1135 – 1144, 2012. ISSN 1877-0428. Proceedings of EWGT2012 - 15th Meeting of the EURO Working Group on Transportation, September 2012, Paris. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1877042812042905>.

ASTARITA, V. et al. New methodology for the identification of road surface anomalies. In: **Proceedings of 2014 IEEE International Conference on Service Operations and Logistics, and Informatics**. [S.l.: s.n.], 2014. p. 149–154.

BADUROWICZ, M.; MONTUSIEWICZ, J.; KARZMAREK, P. Detection of road artefacts using fuzzy adaptive thresholding. In: IEEE. **2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)**. [S.l.], 2020. p. 1–8.

BEEBY, S. **MEMS Mechanical Sensors**. Artech House, 2004. (Artech House Microelectromechanical Systems). ISBN 9781580538732. Disponível em: <https://books.google.com.br/books?id=6wg5oXzks9UC>.

BELLO-SALAU, H. et al. New road anomaly detection and characterization algorithm for autonomous vehicles. **Applied Computing and Informatics**, 2018. ISSN 2210-8327. Disponível em: <http://www.sciencedirect.com/science/article/pii/S2210832718300309>.

BIANCHI, F. et al. **Recurrent Neural Networks for Short-Term Load Forecasting: An Overview and Comparative Analysis**. Springer International Publishing, 2017. (SpringerBriefs in Computer Science). ISBN 9783319703381. Disponível em: <https://books.google.com.br/books?id=wu09DwAAQBAJ>.

BIOLCHINI, J. et al. Systematic review in software engineering. **System Engineering and Computer Science Department COPPE/UFRJ, Technical Report ES**, v. 679, n. 05, p. 45, 2005.

BONAT, D. **Metodologia Da Pesquisa**. Iesde Brasil Sa, 2009. ISBN 9788538703952. Disponível em: <https://books.google.com.br/books?id=5Eesvwncx6sC>.

BOSE, B. et al. Smartphone based system for real-time aggressive driving detection and marking rash driving-prone areas. In: **Proceedings of the Workshop Program of the 19th International Conference on Distributed Computing and Networking**. New York, NY, USA: ACM, 2018. (Workshops ICDCN '18), p. 27:1–27:6. ISBN 978-1-4503-6397-6. Disponível em: <http://doi.acm.org/10.1145/3170521.3170549>.

BRAGA, N. **Manual de Mecatrônica**. Editora NCB, 2017. ISBN 9788565050609. Disponível em: <https://books.google.com.br/books?id=Zr3iBgAAQBAJ>.

BROWNLEE, J. **Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python**. Machine Learning Mastery, 2018. Disponível em: <https://books.google.com.br/books?id=o5qnDwAAQBAJ>.

BROWNLEE, J. **Imbalanced Classification with Python: Better Metrics, Balance Skewed Classes, Cost-Sensitive Learning**. Machine Learning Mastery, 2020. Disponível em: <https://books.google.com.br/books?id=jaXJDwAAQBAJ>.

BRUNAUER, R.; REHRL, K. Supporting road maintenance with in-vehicle data: Results from a field trial on road surface condition monitoring. In: **2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2016. p. 2236–2241. ISSN 2153-0017.

CARLOS, R. et al. Becoming smarter at characterizing potholes and speed bumps from smartphone data - introducing a second-generation inference problem. **IEEE Transactions on Mobile Computing**, p. 1–1, 2019.

CELAYA-PADILLA, J. et al. Speed bump detection using accelerometric features: A genetic algorithm approach. **Sensors**, MDPI AG, v. 18, n. 2, p. 443, Feb 2018. ISSN 1424-8220. Disponível em: <http://dx.doi.org/10.3390/s18020443>.

CHEN, K. et al. Crsm: Crowdsourcing based road surface monitoring. In: **2013 IEEE 10th International Conference on High Performance Computing and Communications 2013 IEEE International Conference on Embedded and Ubiquitous Computing**. [S.l.: s.n.], 2013. p. 2151–2158.

CHEN, K. et al. Crsm: a practical crowdsourcing-based road surface monitoring system. **Wireless Networks**, Springer, v. 22, n. 3, p. 765–779, 2016.

CHEN, Y.; XUE, Y. A deep learning approach to human activity recognition based on single accelerometer. In: **2015 IEEE International Conference on Systems, Man, and Cybernetics**. [S.l.: s.n.], 2015. p. 1488–1492.

CHOUDHARY, A. K.; INGOLE, P. K. Smart phone based approach to monitor driving behavior and sharing of statistic. In: **2014 Fourth International Conference on Communication Systems and Network Technologies**. [S.l.: s.n.], 2014. p. 279–282.

CHRIS, I. **K-Means Clustering Explained Visually In 5 Minutes**. 2020. Disponível em: <https://medium.com/dataseries/k-means-clustering-explained-visually-in-5-minutes-b900cc69d175>.

CNT. **Pesquisa CNT de Rodovias 2017: Relatório Gerencial**. Brasília: CNT, SEST, SENAT. 403 p. Brasília: [s.n.], 2017.

DEEP, S.; ZHENG, X. Hybrid model featuring cnn and lstm architecture for human activity recognition on smartphone sensor data. In: **2019 20th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)**. [S.l.: s.n.], 2019. p. 259–264.

DENG, L.; YU, D. et al. Deep learning: methods and applications. **Foundations and Trends® in Signal Processing**, Now Publishers, Inc., v. 7, n. 3–4, p. 197–387, 2014.

DEY, M. R. et al. Magtrack: Detecting road surface condition using smartphone sensors and machine learning. In: **TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)**. [S.l.: s.n.], 2019. p. 2485–2489. ISSN 2159-3450.

DIXON, P. et al. Machine learning algorithms can classify outdoor terrain types during running using accelerometry data. **Gait and Posture**, v. 74, p. 176 – 181, 2019. ISSN 0966-6362. Disponible em: <http://www.sciencedirect.com/science/article/pii/S0966636219307064>.

DOUANGPHACHANH, V.; ONEYAMA, H. Estimation of road roughness condition from smartphones under realistic settings. In: **2013 13th International Conference on ITS Telecommunications (ITST)**. [S.l.: s.n.], 2013. p. 433–439.

DOUANGPHACHANH, V.; ONEYAMA, H. Exploring the use of smartphone accelerometer and gyroscope to study on the estimation of road surface roughness condition. In: **2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO)**. [S.l.: s.n.], 2014. v. 01, p. 783–787.

DOUANGPHACHANH, V.; ONEYAMA, H. A study on the use of smartphones under realistic settings to estimate road roughness condition. **EURASIP Journal on Wireless Communications and Networking**, Springer, v. 2014, n. 1, p. 1–11, 2014.

DU, Y. et al. Measurement of international roughness index by using z-axis accelerometers and gps. **Mathematical Problems in Engineering**, Hindawi Publishing Corporation, v. 2014, p. 928980, Jun 2014. ISSN 1024-123X. Disponible em: <https://doi.org/10.1155/2014/928980>.

EDWAN, E.; SARSOUR, N.; ALATRASH, M. Mobile application for bumps detection and warning utilizing smartphone sensors. In: **2019 International Conference on Promising Electronic Technologies (ICPET)**. [S.l.: s.n.], 2019. p. 50–54.

EFTEKHARI, H. R.; GHATEE, M. Hybrid of discrete wavelet transform and adaptive neuro fuzzy inference system for overall driving behavior recognition. **Transportation Research Part F: Traffic Psychology and Behaviour**, v. 58, p. 782 – 796, 2018. ISSN 1369-8478. Disponible em: <http://www.sciencedirect.com/science/article/pii/S1369847817306216>.

EL-WAKEEL, A. S. et al. Towards a practical crowdsensing system for road surface conditions monitoring. **IEEE Internet of Things Journal**, v. 5, n. 6, p. 4672–4685, Dec 2018. ISSN 2327-4662.

FERNÁNDEZ, A. et al. **Learning from imbalanced data sets**. [S.l.]: Springer, 2018.

FOLEY, D. **K-Means Clustering**. Towards Data Science, 2019. Disponible em: <https://towardsdatascience.com/k-means-clustering-8e1e64c1561c>.

FOUAD, M. M. et al. Intelligent road surface quality evaluation using rough mereology. In: **2014 14th International Conference on Hybrid Intelligent Systems**. [S.l.: s.n.], 2014. p. 18–22.

GAWAD, S. M. A.; MOUGY, A. E.; EL-MELIGY, M. A. Dynamic mapping of road conditions using smartphone sensors and machine learning techniques. In: **2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)**. [S.l.: s.n.], 2016. p. 1–5.

GEYER, J. et al. A2d2: Audi autonomous driving dataset. **arXiv preprint arXiv:2004.06320**, 2020.

GHADGE, M.; PANDEY, D.; KALBANDE, D. Machine learning approach for predicting bumps on road. In: **2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATcCT)**. [S.l.: s.n.], 2015. p. 481–485.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. (Adaptive Computation and Machine Learning series). ISBN 9780262035613. Disponível em: <https://books.google.com.br/books?id=Np9SDQAAQBAJ>.

GROVES, P. **Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition**. Artech House, 2013. (GNSS/GPS). ISBN 9781608070053. Disponível em: <https://books.google.com.br/books?id=t94fAgAAQBAJ>.

GUETA, L. B.; SATO, A. Classifying road surface conditions using vibration signals. In: **2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)**. [S.l.: s.n.], 2017. p. 039–043.

GUNAWAN, F. E.; YANFI; SOEWITO, B. A vibratory-based method for road damage classification. In: **2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)**. [S.l.: s.n.], 2015. p. 1–4.

GăVRUTă, A.; MARCU, M.; BOGDAN, R. Software solution for monitoring and analyzing driver behavior. In: **2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)**. [S.l.: s.n.], 2018. p. 000345–000350.

HASSAN, N. et al. Road anomaly classification for low-cost road maintenance and route quality maps. In: IEEE. **2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)**. [S.l.], 2019. p. 645–650.

HE, H.; MA, Y. **Imbalanced Learning: Foundations, Algorithms, and Applications**. Wiley, 2013. ISBN 9781118646335. Disponível em: <https://books.google.com.br/books?id=CVHx-Gp9jzUC>.

HOU, Y. et al. Vehsense: Slippery road detection using smartphones. In: **2017 IEEE 85th Vehicular Technology Conference (VTC Spring)**. [S.l.: s.n.], 2017. p. 1–5.

HSU, Y.; PERNG, J.; WU, Z. Design and implementation of an intelligent road detection system with multisensor integration. In: **2016 International Conference on Machine Learning and Cybernetics (ICMLC)**. [S.l.: s.n.], 2016. v. 1, p. 219–225. ISSN 2160-1348.

IDRIS, M. K.; BOUKAR, M. M.; ADESHINA, S. A. Analysis of bad roads using smart phone. In: **2019 15th International Conference on Electronics, Computer and Computation (ICECCO)**. [S.l.: s.n.], 2019. p. 1–4.

JAMAKHANDI, H. A.; SRINIVASA, K. G. Internet of things based real time mapping of road irregularities. In: **International Conference on Circuits, Communication, Control and Computing**. [S.l.: s.n.], 2014. p. 448–451.

JANG, J. et al. Road surface condition monitoring via multiple sensor-equipped vehicles. In: **2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)**. [S.l.: s.n.], 2015. p. 43–44.

JONES, M. T. **Arquiteturas de aprendizado profundo: O surgimento da inteligência artificial**. IBM, 2017. Disponível em: <https://www.ibm.com/developerworks/br/library/cc-machine-learning-deep-learning-architectures/index.html>.

JOSé, I. **KNN (K-Nearest Neighbors) #1**. 2018. Disponível em: <https://medium.com/brasil-ai/knn-k-nearest-neighbors-1-e140c82e9c4e>.

- JUNIOR, J. R. F. **Redes Neurais Recorrentes - LSTM**. Medium, 2019. Disponível em: <https://medium.com/@web2ajax/redes-neurais-recorrentes-lstm-b90b720dc3f6>.
- KAUR, V. et al. Crowd-sourcing based android application for structural health monitoring and data analytics of roads using cloud computing. In: **2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)**. [S.l.: s.n.], 2017. p. 354–358.
- KEMPE, V. **Inertial MEMS: Principles and Practice**. Cambridge University Press, 2011. ISBN 9781139494823. Disponível em: <https://books.google.com.br/books?id=XzdvdGblZ8EC>.
- KHALEGHIAN, S.; TAHERI, S. Terrain classification using intelligent tire. **Journal of Terramechanics**, v. 71, p. 15 – 24, 2017. ISSN 0022-4898. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0022489817300125>.
- KHANDELWAL, R. **K-Nearest Neighbors (KNN)**. 2018. Disponível em: <https://medium.com/datadriveninvestor/k-nearest-neighbors-knn-7b4bd0128da7>.
- KITCHENHAM, B. Procedures for performing systematic reviews. **Keele, UK, Keele University**, v. 33, n. 2004, p. 1–26, 2004.
- KITCHENHAM, B. et al. Systematic literature reviews in software engineering – a systematic literature review. **Information and Software Technology**, v. 51, n. 1, p. 7 – 15, 2009. ISSN 0950-5849. Special Section - Most Cited Articles in 2002 and Regular Research Papers. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0950584908001390>.
- KRAWCZYK, B. Learning from imbalanced data: open challenges and future directions. **Progress in Artificial Intelligence**, Springer, v. 5, n. 4, p. 221–232, 2016.
- KRIG, S. Ground truth data, content, metrics, and analysis. In: \_\_\_\_\_. **Computer Vision Metrics: Survey, Taxonomy, and Analysis**. Berkeley, CA: Apress, 2014. p. 283–311. ISBN 978-1-4302-5930-5. Disponível em: [https://doi.org/10.1007/978-1-4302-5930-5\\_7](https://doi.org/10.1007/978-1-4302-5930-5_7).
- KUHN, M.; JOHNSON, K. et al. **Applied predictive modeling**. [S.l.]: Springer, 2013. v. 26.
- KUMAR, G. A. et al. Road quality management system using mobile sensors. In: **2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)**. [S.l.: s.n.], 2017. p. 1–6.
- KUMAR, R.; WIL, U. **Recent Advances in Computational Intelligence**. [S.l.]: Springer International Publishing, 2019. (Studies in Computational Intelligence). ISBN 9783030125004.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015.
- LEI, T.; MOHAMED, A. A.; CLAUDEL, C. An imu-based traffic and road condition monitoring system. **HardwareX**, Elsevier, v. 4, p. e00045, 2018.
- LEIZEROVYCH, R. et al. Iot-complex for monitoring and analysis of motor highway condition using artificial neural networks. In: IEEE. **2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)**. [S.l.], 2020. p. 207–212.
- LI, F. et al. Dangerous driving behavior detection using smartphone sensors. In: **2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2016. p. 1902–1907. ISSN 2153-0017.

LI, X.; GOLDBERG, D. W. Toward a mobile crowdsensing system for road surface assessment. **Computers, Environment and Urban Systems**, v. 69, p. 51 – 62, 2018. ISSN 0198-9715. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0198971517301333>.

LIMA, L. C. et al. Using crowdsourcing techniques and mobile devices for asphaltic pavement quality recognition. In: **2016 VI Brazilian Symposium on Computing Systems Engineering (SBESC)**. [S.l.: s.n.], 2016. p. 144–149. ISSN 2324-7894.

LOIZOS, A.; PLATI, C. Evolutional process of pavement roughness evaluation benefiting from sensor technology. **International Journal on Smart Sensing and Intelligent Systems**, v. 1, 01 2008.

M., H. P.; GOPI, V. P. Vehicle vibration signal processing for road surface monitoring. **IEEE Sensors Journal**, v. 17, n. 16, p. 5192–5197, Aug 2017. ISSN 1530-437X.

MAHBOOB, S. et al. Drive pattern analyzer using smartphones. In: **2017 13th International Conference on Emerging Technologies (ICET)**. [S.l.: s.n.], 2017. p. 1–5.

MASINO, J. et al. Learning from the crowd: Road infrastructure monitoring system. **Journal of Traffic and Transportation Engineering (English Edition)**, v. 4, n. 5, p. 451 – 463, 2017. ISSN 2095-7564. Disponível em: <http://www.sciencedirect.com/science/article/pii/S2095756417300168>.

MATHEW, T. V. **Transportation Systems Engineering: Intrusive Technologies**. IIT Bombay, 2014. Disponível em: [https://nptel.ac.in/courses/105101008/downloads/cete\\_09.pdf](https://nptel.ac.in/courses/105101008/downloads/cete_09.pdf).

MATHEW, T. V. **Transportation Systems Engineering: Non-Intrusive Technologies**. IIT Bombay, 2014. Disponível em: [https://nptel.ac.in/courses/105101008/downloads/cete\\_10.pdf](https://nptel.ac.in/courses/105101008/downloads/cete_10.pdf).

MATILAINEN, M.; TUONONEN, A. Tyre contact length on dry and wet road surfaces measured by three-axial accelerometer. **Mechanical Systems and Signal Processing**, v. 52-53, p. 548 – 558, 2015. ISSN 0888-3270. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0888327014003185>.

MENEGAZZO, J.; WANGENHEIM, A. V. Vehicular perception and proprioception based on inertial sensing: a systematic review. **Relatórios Técnicos do INCoD**, v. 8, p. 1 – 89, 10 2018. ISSN 2236-5281. Disponível em: <http://www.incod.ufsc.br/wp-content/uploads/2018/10/INCoD-TR-2018-07-LAPIX-E-V01.pdf>.

MENEGAZZO, J.; WANGENHEIM, A. von. Vehicular perception based on inertial sensing: A structured mapping of approaches and methods. **SN Computer Science**, p. 275, Aug 2020. ISSN 2661-8907. Disponível em: <https://doi.org/10.1007/s42979-020-00275-z>.

MEOCCI, M.; BRANZI, V.; SANGIOVANNI, A. An innovative approach for high-performance road pavement monitoring using black box. **Journal of Civil Structural Health Monitoring**, Springer, p. 1–22, 2021.

MILETTE, G.; STROUD, A. **Professional Android Sensor Programming**. Wiley, 2012. (Professional Android Sensor Programming). ISBN 9781118183489. Disponível em: <https://books.google.com.br/books?id=QZaIIIDmoWb8C>.

MORETTIN, P.; TOLOI, C. **Análise de séries temporais: modelos lineares univariados**. Blucher, 2018. ISBN 9788521213529. Disponível em: <https://books.google.com.br/books?id=UwC5DwAAQBAJ>.

- MÜLLER, M. **Information Retrieval for Music and Motion**. Springer Berlin Heidelberg, 2007. ISBN 9783540740483. Disponível em: <https://books.google.com.br/books?id=kSzeZWR2yDsC>.
- NALAVDE, R.; SURVE, A. Driver assistant services using ubiquitous smartphone. In: **2015 International Conference on Communications and Signal Processing (ICCSP)**. [S.l.: s.n.], 2015. p. 0430–0434.
- NI, D. Chapter 1 - traffic sensing technologies. In: NI, D. (Ed.). **Traffic Flow Theory**. Butterworth-Heinemann, 2016. p. 3 – 17. ISBN 978-0-12-804134-5. Disponível em: <http://www.sciencedirect.com/science/article/pii/B9780128041345000015>.
- NISBET, R.; ELDER, J.; MINER, G. **Handbook of Statistical Analysis and Data Mining Applications**. Elsevier Science, 2009. ISBN 9780080912035. Disponível em: <https://books.google.com.br/books?id=U5np34a5fmQC>.
- NISKANEN, A. J.; TUONONEN, A. J. Accelerometer tyre to estimate the aquaplaning state of the tyre-road contact. In: **2015 IEEE Intelligent Vehicles Symposium (IV)**. [S.l.: s.n.], 2015. p. 343–348. ISSN 1931-0587.
- NUNES, D. E.; MOTA, V. F. A participatory sensing framework to classify road surface quality. **Journal of Internet Services and Applications**, SpringerOpen, v. 10, n. 1, p. 1–16, 2019.
- ORHAN, F.; EREN, P. E. Road hazard detection and sharing with multimodal sensor analysis on smartphones. In: **2013 Seventh International Conference on Next Generation Mobile Apps, Services and Technologies**. [S.l.: s.n.], 2013. p. 56–61. ISSN 2161-2889.
- PATIL, S. O.; VARIYAR., V. V. S.; SOMAN, K. P. Speed bump segmentation an application of conditional generative adversarial network for self-driving vehicles. In: **2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)**. [S.l.: s.n.], 2020. p. 935–939.
- PHI, M. **Illustrated Guide to LSTM's and GRU's: A step by step explanation**. 2020. Disponível em: <https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>.
- PHOLPRASIT, T.; CHOOCHAIWATTANA, W.; SAIPRASERT, C. A comparison of driving behaviour prediction algorithm using multi-sensory data on a smartphone. In: **2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)**. [S.l.: s.n.], 2015. p. 1–6.
- PITOŇÁK, M.; FILIPOVSKY, J. Gpr application – non-destructive technology for verification of thicknesses of newly paved roads in slovakia. **Procedia Engineering**, v. 153, p. 537 – 549, 2016. ISSN 1877-7058. XXV Polish – Russian – Slovak Seminar Theoretical Foundation of Civil Engineering. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1877705816323323>.
- PONT, E. M.; PROVOST, J.; KUENZEL, C. Development of a methodology for monitoring and prediction of road surface conditions in highly automated driving. In: **IEEE. 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)**. [S.l.], 2017. p. 1–7.



POOJA, P. R.; HARIHARAN, B. An early warning system for traffic and road safety hazards using collaborative crowd sourcing. In: **2017 International Conference on Communication and Signal Processing (ICCS)**. [S.l.: s.n.], 2017. p. 1203–1206.

POPPER, K. R. **The Logic of Scientific Discovery**. [S.l.]: Routledge, 2002. 1st English Edition:1959. ISBN 0-415-27844-9.

PRAPULLA, S. B.; RAO, S. N.; HERUR, V. A. Road quality analysis and mapping for faster and safer travel. In: **2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)**. [S.l.: s.n.], 2017. p. 2487–2490.

PUPALE, R. **Support Vector Machines(SVM) — An Overview**. 2019. Disponível em: <https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989>.

RAHMAN, S. A.; ADJEROH, D. A. Deep learning using convolutional lstm estimates biological age from physical activity. **Scientific reports**, Nature Publishing Group, v. 9, n. 1, p. 1–15, 2019.

RISHIWAL, V.; KHAN, H. Automatic pothole and speed breaker detection using android system. In: **2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)**. [S.l.: s.n.], 2016. p. 1270–1273.

RODRIGUES, V. **Métricas de Avaliação: acurácia, precisão, recall... quais as diferenças?** 2020. Disponível em: <https://medium.com/@vitorborbarodrigues/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-recall-quais-as-diferen%C3%A7as-c8f05e0a513c>.

RODRIGUES, V. **Machine Learning: o que são Accuracy, Precision, Recall e F1 Score**. 2021. Disponível em: <https://medium.com/@wilsonrodrigues/machine-learning-o-que-s%C3%A3o-accuracy-precision-recall-e-f1-score-f16762f165b0>.

SABIR, N.; MEMON, A. A.; SHAIKH, F. K. Threshold based efficient road monitoring system using crowdsourcing approach. **Wireless Personal Communications**, Springer, v. 106, n. 4, p. 2407–2425, 2019.

SAIPRASERT, C. et al. Driver behaviour profiling using smartphone sensory data in a v2i environment. In: **2014 International Conference on Connected Vehicles and Expo (ICCVE)**. [S.l.: s.n.], 2014. p. 552–557. ISSN 2378-1289.

SALEH, K.; HOSSNY, M.; NAHAVANDI, S. Driving behavior classification based on sensor data fusion using lstm recurrent neural networks. In: **2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2017. p. 1–6. ISSN 2153-0017.

SALMAN, A. G. et al. Single layer & multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting. **Procedia Computer Science**, v. 135, p. 89 – 98, 2018. ISSN 1877-0509. The 3rd International Conference on Computer Science and Computational Intelligence (ICCS 2018) : Empowering Smart Technology in Digital Era for a Better Life. Disponível em: <http://www.sciencedirect.com/science/article/pii/S187705091831439X>.

SATTAR, S.; LI, S.; CHAPMAN, M. Road surface monitoring using smartphone sensors: A review. **Sensors**, MDPI AG, v. 18, n. 11, p. 3845, Nov 2018. ISSN 1424-8220. Disponível em: <http://dx.doi.org/10.3390/s18113845>.

SAVERA, A. et al. Bumpster: A mobile cloud computing system for speed breakers and ditches. In: **2016 IEEE 41st Conference on Local Computer Networks Workshops (LCN Workshops)**. [S.l.: s.n.], 2016. p. 65–71.

SCIKIT-LEARN. **Scikit-Learn Cluster: KMeans**. Scikit-Learn, 2021. Disponível em: <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>. Acesso em: 20 jun. de 2021.

SCIKIT-LEARN. **Scikit-Learn Neighbors: KNeighborsClassifier**. Scikit-Learn, 2021. Disponível em: <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>. Acesso em: 20 jun. de 2021.

SCIKIT-LEARN. **Scikit-Learn Support Vector Machine: SVC**. Scikit-Learn, 2021. Disponível em: <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>. Acesso em: 20 jun. de 2021.

SEBASTIAN, B.; BEN-TZVI, P. Support vector machine based real-time terrain estimation for tracked robots. **Mechatronics**, v. 62, p. 102260, 2019. ISSN 0957-4158. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957415819300935>.

SELMANAJ, D.; CORNO, M.; SAVARESI, S. M. Accelerometer-based data-driven hazard detection and classification for motorcycles. In: **2014 European Control Conference (ECC)**. [S.l.: s.n.], 2014. p. 1687–1692.

SHARMA, H. et al. S-road assist: Road surface conditions and driving behavior analysis using smartphones. In: **2015 International Conference on Connected Vehicles and Expo (ICCVE)**. [S.l.: s.n.], 2015. p. 291–296. ISSN 2378-1297.

SHRESTHA, A.; WON, M. Deepwalking: Enabling smartphone-based walking speed estimation using deep learning. In: **2018 IEEE Global Communications Conference (GLOBECOM)**. [S.l.: s.n.], 2018. p. 1–6.

SHUBHAM, J. **SUPPORT VECTOR MACHINES (SVM)**. 2018. Disponível em: <https://medium.com/coinmonks/support-vector-machines-svm-b2b433419d73>.

SHUNG, K. P. **Accuracy, Precision, Recall or F1?** 2020. Disponível em: <https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9>.

SINGH, B.; GUPTA, A. Recent trends in intelligent transportation systems: a review. **Journal of Transport Literature**, scielo, v. 9, p. 30 – 34, 04 2015. ISSN 2238-1031.

SINGH, G. et al. Smart patrolling: An efficient road surface monitoring using smartphone sensors and crowdsourcing. **Pervasive and Mobile Computing**, v. 40, p. 71 – 88, 2017. ISSN 1574-1192. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1574119216301262>.

SINGH, V. et al. Safestreet: An automated road anomaly detection and early-warning system using mobile crowdsensing. In: **2018 10th International Conference on Communication Systems Networks (COMSNETS)**. [S.l.: s.n.], 2018. p. 549–552. ISSN 2155-2509.

SOUZA, V. M. Asphalt pavement classification using smartphone accelerometer and complexity invariant distance. **Engineering Applications of Artificial Intelligence**, v. 74, p. 198 – 211, 2018. ISSN 0952-1976. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0952197618301349>.

SOUZA, V. M.; GIUSTI, R.; BATISTA, A. J. Asfalt: A low-cost system to evaluate pavement conditions in real-time using smartphones and machine learning. **Pervasive and Mobile Computing**, Elsevier, v. 51, p. 121–137, 2018.

SRIMONGKON, S.; CHIRACHARIT, W. Detection of speed bumps using gaussian mixture model. In: **2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)**. [S.l.: s.n.], 2017. p. 628–631.

STRUTU, M.; STAMATESCU, G.; POPESCU, D. A mobile sensor network based road surface monitoring system. In: **2013 17th International Conference on System Theory, Control and Computing (ICSTCC)**. [S.l.: s.n.], 2013. p. 630–634.

TAN, H. X. et al. Time series classification using a modified lstm approach from accelerometer-based data: A comparative study for gait cycle detection. **Gait and Posture**, v. 74, p. 128 – 134, 2019. ISSN 0966-6362.

TENSORFLOW. **TensorFlow Keras Layers: Conv1D**. TensorFlow, 2021. Disponível em: [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/Conv1D](https://www.tensorflow.org/api_docs/python/tf/keras/layers/Conv1D). Acesso em: 20 jun. de 2021.

TENSORFLOW. **TensorFlow Keras Layers: ConvLSTM2**. TensorFlow, 2021. Disponível em: [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/ConvLSTM2](https://www.tensorflow.org/api_docs/python/tf/keras/layers/ConvLSTM2). Acesso em: 20 jun. de 2021.

TENSORFLOW. **TensorFlow Keras Layers: GRU**. TensorFlow, 2021. Disponível em: [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/GRU](https://www.tensorflow.org/api_docs/python/tf/keras/layers/GRU). Acesso em: 20 jun. de 2021.

TENSORFLOW. **TensorFlow Keras Layers: LSTM**. TensorFlow, 2021. Disponível em: [https://www.tensorflow.org/api\\_docs/python/tf/keras/layers/LSTM](https://www.tensorflow.org/api_docs/python/tf/keras/layers/LSTM). Acesso em: 20 jun. de 2021.

TIWARI, S.; BHANDARI, R.; RAMAN, B. Roadcare: a deep-learning based approach to quantifying road surface quality. In: **Proceedings of the 3rd ACM SIGCAS Conference on Computing and Sustainable Societies**. [S.l.: s.n.], 2020. p. 231–242.

TOLENTINO-RABELO, R.; MUÑOZ, D. M. Online terrain classification for mobile robots using fpgas. In: **2016 IEEE 7th Latin American Symposium on Circuits Systems (LASCAS)**. [S.l.: s.n.], 2016. p. 231–234.

TOMIYAMA, K.; KAWAMURA, A. Application of lifting wavelet transform for pavement surface monitoring by use of a mobile profilometer. **International Journal of Pavement Research and Technology**, v. 9, n. 5, p. 345 – 353, 2016. ISSN 1996-6814. Disponível em: <http://www.sciencedirect.com/science/article/pii/S199668141630075X>.

VAITHEESHWARI, R.; SATHIESHKUMAR, V. Performance analysis of epileptic seizure detection system using neural network approach. In: **2019 International Conference on Computational Intelligence in Data Science (ICCIDS)**. [S.l.: s.n.], 2019. p. 1–5.

VERMA, R. et al. Urbaneye: An outdoor localization system for public transport. In: **IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications**. [S.l.: s.n.], 2016. p. 1–9.

VERMA, S. **Understanding 1D and 3D Convolution Neural Network | Keras**. 2020. Disponível em: <https://towardsdatascience.com/understanding-1d-and-3d-convolution-neural-network-keras-9d8f76e29610>.

VITTORIO, A. et al. Automated sensing system for monitoring of road surface quality by mobile devices. **Procedia - Social and Behavioral Sciences**, v. 111, p. 242 – 251, 2014. ISSN 1877-0428. Transportation: Can we do more with less resources? – 16th Meeting of the Euro Working Group on Transportation – Porto 2013. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1877042814000585>.

VLAHOGIANNI, E. I.; BARMPOUNAKIS, E. N. Driving analytics using smartphones: Algorithms, comparisons and challenges. **Transportation Research Part C: Emerging Technologies**, v. 79, p. 196 – 206, 2017. ISSN 0968-090X. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0968090X17300980>.

WANG, H. et al. A road quality detection method based on the mahalanobis-taguchi system. **IEEE Access**, v. 6, p. 29078–29087, 2018. ISSN 2169-3536.

WANG, J. et al. Deep learning for sensor-based activity recognition: A survey. **Pattern Recognition Letters**, v. 119, p. 3 – 11, 2019. ISSN 0167-8655. Deep Learning for Pattern Recognition. Disponível em: <http://www.sciencedirect.com/science/article/pii/S016786551830045X>.

WANG, S. et al. Road-terrain classification for land vehicles: Employing an acceleration-based approach. **IEEE Vehicular Technology Magazine**, v. 12, n. 3, p. 34–41, 2017.

WANG, S. et al. Two-stage road terrain identification approach for land vehicles using feature-based and markov random field algorithm. **IEEE Intelligent Systems**, v. 33, n. 1, p. 29–39, 2018.

WICKRAMARATHNE, T.; GARG, V.; BAUER, P. On the use of 3-d accelerometers for road quality assessment. In: **2018 IEEE 87th Vehicular Technology Conference (VTC Spring)**. [S.l.: s.n.], 2018. p. 1–5. ISSN 2577-2465.

WU, B. et al. Reasoning-based framework for driving safety monitoring using driving event recognition. **IEEE Transactions on Intelligent Transportation Systems**, v. 14, n. 3, p. 1231–1241, Sep. 2013. ISSN 1524-9050.

WU, X. et al. Abnormal driving behavior detection for bus based on the bayesian classifier. In: **2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)**. [S.l.: s.n.], 2018. p. 266–272.

YANG, J. et al. Smartphones based online activity recognition for indoor localization using deep convolutional neural network. In: **2018 Ubiquitous Positioning, Indoor Navigation and Location-Based Services (UPINLBS)**. [S.l.: s.n.], 2018. p. 1–7.

ZACCONE, G.; KARIM, M.; MENSRAWY, A. **Deep Learning with TensorFlow**. Packt Publishing, 2017. ISBN 9781786460127. Disponível em: <https://books.google.com.br/books?id=C0IwDwAAQBAJ>.

ZEBIN, T. et al. Implementation of a batch normalized deep lstm recurrent network on a smartphone for human activity recognition. In: **2019 IEEE EMBS International Conference on Biomedical Health Informatics (BHI)**. [S.l.: s.n.], 2019. p. 1–4.

ZEBIN, T. et al. Human activity recognition from inertial sensor time-series using batch normalized deep lstm recurrent networks. In: **2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)**. [S.l.: s.n.], 2018. p. 1–4.

ZHANG, J. et al. Data-driven intelligent transportation systems: A survey. **IEEE Transactions on Intelligent Transportation Systems**, v. 12, n. 4, p. 1624–1639, 2011.

ZHAO, B. et al. Iri estimation by the frequency domain analysis of vehicle dynamic responses and its large-scale application. In: **Adjunct Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing Networking and Services**. [S.l.: s.n.], 2016. p. 41–46.