

Aplicação de Inteligência Artificial na Estimação de Estados em Veículos Inteligentes

Jhullye Freitas Rodrigues
Departamento de Automática
Universidade Federal de Lavras
Lavras, MG, Brasil
jhullye.rodrigues1@estudante.ufla.br

Rafael Ávila dos Santos
Programa de Pós-Graduação em
Engenharia de Sistemas e Automação
Universidade Federal de Lavras
Lavras, MG, Brasil
rafael.santos0t5@gmail.com

Bruno H. G. Barbosa
Departamento de Automática
Universidade Federal de Lavras
Lavras, MG, Brasil
brunohb@ufla.br

André Murilo de Almeida Pinto
Departamento de Automática
Universidade Federal de Lavras
Lavras, MG, Brasil
andremurilo@ufla.br

Henrique Carvalho de Castro
Programa de Pós-Graduação em
Engenharia de Sistemas e Automação
Universidade Federal de Lavras
Lavras, MG, Brasil
henriquec.castro@outlook.com

Nan Xu
National Key Laboratory of
Automotive Chassis Integration
Jilin University
Changchun, JL, China
xunan@jlu.edu.cn

Abstract—A estimação das velocidades longitudinal (v_x) e lateral (v_y) é fundamental para a estabilidade e segurança dos veículos inteligentes, permitindo também o cálculo do ângulo de deriva. No entanto, métodos tradicionais baseados em modelos físicos são complexos e pouco práticos em cenários dinâmicos. Este trabalho propõe a utilização de redes neurais recorrentes, *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU), para estimar essas velocidades a partir de dados experimentais de um veículo inteligente. Os modelos foram treinados e validados com cinco conjuntos de dados obtidos de um veículo de passeio conduzido em situações extremas de segurança. Ambos apresentaram resultados satisfatórios para predição da velocidade longitudinal, mas com maior dificuldade na predição da velocidade lateral em situações mais complexas, com a GRU destacando-se pela eficiência computacional e a LSTM pelo desempenho. As redes recorrentes mostraram-se alternativas viáveis para substituir sensores físicos, de forma a possibilitar o desenvolvimento de *soft-sensors*, promovendo sistemas mais acessíveis na mobilidade inteligente.

Index Terms—Dinâmica Veicular, Pneu Inteligente, Machine Learning, Inteligência Computacional, Sistemas Inteligentes.

I. INTRODUÇÃO

Os veículos inteligentes emergiram na década de 1970, com a implementação dos primeiros sistemas eletrônicos, como o sistema *Anti-lock Braking System* (ABS) e os controles de injeção eletrônica. Já durante as décadas de 1980 e 1990, o desenvolvimento de sensores e processadores experimentou um avanço significativo, resultando na criação de recursos como o controle de tração, a suspensão ativa e os primeiros sistemas de navegação [1].

A estimação precisa das velocidades longitudinais e laterais desempenha papel crucial em termos de segurança, estabilidade e eficiência operacional. Essas velocidades são fundamentais para o controle efetivo e seguro dos veículos, especialmente em condições críticas como curvas acentuadas,

mudanças repentinas de direção e situações de emergência. Além disso, a determinação precisa dessas velocidades permite calcular o ângulo de deriva, um parâmetro crítico para avaliar a estabilidade direcional, prevenir acidentes e melhorar o desempenho dinâmico [2]. Métodos baseados em modelos físicos enfrentam limitações, pois demandam uma modelagem detalhada do comportamento do veículo, levando em conta fatores como atrito dos pneus e efeitos aerodinâmicos. Essas abordagens resultam em equações matemáticas complexas e muitas vezes pouco práticas em cenários dinâmicos [3].

A Inteligência Artificial (IA) surge como uma solução promissora para superar essas barreiras. A partir do início do século XXI, a IA tornou-se uma aliada essencial no desenvolvimento dos veículos inteligentes. O crescente uso de IA tem desempenhado um papel central no avanço dos veículos inteligentes, permitindo a integração de algoritmos capazes de processar dados de sensores em tempo real. Isso possibilita a estimação de estados dinâmicos críticos, como velocidades e ângulos de deriva, aumentando a segurança e eficiência dos sistemas [1].

Algoritmos de IA como Redes Neurais Artificiais (RNAs), em especial as arquiteturas avançadas como Redes Neurais Recorrentes (RNNs), *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU), destacam-se por sua capacidade de aprender padrões complexos diretamente dos dados, eliminando a necessidade de modelagem física explícita. Além disso, essas redes permitem o uso de sensores mais simples, como acelerômetros e giroscópios, reduzindo custos e ampliando a acessibilidade da tecnologia [4].

Estudos recentes mostram que RNNs podem igualar ou até superar os resultados obtidos com sensores de alto custo, consolidando-se como uma alternativa viável para aplicações comerciais e industriais em veículos inteligentes [5]. Essa

capacidade de combinar eficiência e baixo custo torna a IA uma ferramenta essencial para a mobilidade do futuro, oferecendo soluções inovadoras para desafios como condições climáticas extremas, mudanças rápidas no terreno e tráfego imprevisível. Essa abordagem proporciona maior precisão nas previsões e permite a substituição de sensores físicos por sensores virtuais [6].

Com base nesse contexto, este trabalho propõe explorar o potencial de diferentes arquiteturas de redes neurais recorrentes na estimação da velocidade longitudinal (v_x) e lateral (v_y) utilizando dados experimentais de um veículo inteligente. O objetivo é identificar soluções que aliam precisão, robustez e eficiência, contribuindo para o avanço da tecnologia de mobilidade inteligente.

Este artigo está estruturado da seguinte forma: a Seção II descreve o modelo veicular utilizado, assim como as redes neurais utilizadas. Na Seção III é visto o banco de dados, assim como o pré-processamento realizado e as configurações estabelecidas para as redes neurais. A Seção IV mostra os resultados obtidos e as discussões. Por fim, a Seção V apresenta as conclusões.

II. REFERENCIAL TEÓRICO

Este capítulo apresenta a base teórica necessária para o entendimento e desenvolvimento deste trabalho, abordando os conceitos de modelagem de veículos e as principais técnicas de IA, com foco nas redes neurais recorrentes LSTM e GRU.

A. Modelo Bicicleta para Modelagem Veicular

A modelagem cinemática constitui uma ferramenta de caráter essencial para a descrição do movimento de veículos em um plano bidimensional. A partir desta, é possível correlacionar variáveis de entrada, tais como acelerações e velocidades das rodas, com os estados dinâmicos do veículo, designadamente a sua velocidade longitudinal (v_x) e lateral (v_y) [7]. No âmbito do presente estudo, foi adotado o Modelo Bicicleta, o qual simplifica a dinâmica veicular ao considerar dois pontos de contato com o solo, um no eixo dianteiro e outro no eixo traseiro. Consequentemente, no Modelo Bicicleta, é obtida a média da leitura das grandezas físicas atuantes nas rodas frontais e traseiras do veículo, conforme apresentado na Figura 1.

O Modelo de Bicicleta utiliza o Centro de Gravidade (CG) como ponto de referência para a análise do movimento. O efeito combinado das forças longitudinais e laterais, bem como a taxa de guinada (Ψ), é considerado. Esta última descreve a rotação do veículo em torno do seu eixo vertical (y). Em momentos em que ocorre rotação das rodas a fim de deslocar o veículo lateralmente, existirá o ângulo de deslizamento das rodas traseiras (α_r) e dianteiras (α_f), assim como o ângulo de deslizamento do veículo (β).

A velocidade longitudinal está associada ao deslocamento na direção do eixo do veículo. A equação 1 indica que a variação da velocidade longitudinal ($\dot{v}_x$) depende da aceleração nesta direção e do efeito rotacional combinado com a velocidade lateral [8]:

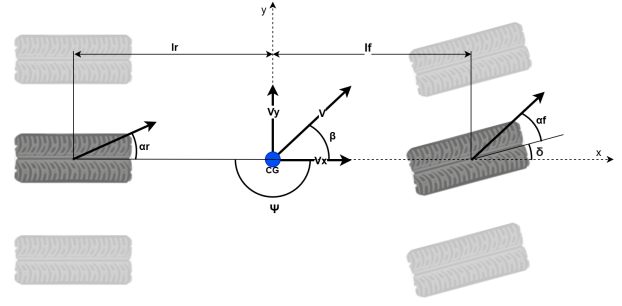


Fig. 1. Modelo de Bicicleta para Cinética Veicular.

$$\dot{v}_x = a_x + \Psi v_y. \quad (1)$$

A velocidade lateral representa o desvio perpendicular ao eixo, geralmente resultante de manobras ou forças laterais. O equacionamento 2 relaciona a variação da velocidade lateral ($\dot{v}_y$) à aceleração lateral e à interação entre a rotação do veículo e sua velocidade longitudinal [8]:

$$\dot{v}_y = a_y - \Psi v_x. \quad (2)$$

As Equações 1 e 2 expressam como essas velocidades variam ao longo do tempo, sendo influenciadas pelas acelerações correspondentes (a_x , a_y) e pela taxa de guinada (Ψ) [8].

As aplicações do Modelo Bicicleta são amplamente utilizadas para modelagem veicular, servindo como base para diversos algoritmos de controle, incluindo sistemas avançados de assistência ao motorista (ADAS). Ele também é utilizado em simulações para validar estimadores de estados baseados em sensores virtuais, como redes neurais [9].

B. Redes Neurais Recorrentes - LSTM e GRU

As Redes Neurais Recorrentes (RNN) foram desenvolvidas para lidar com dados sequenciais, aproveitando conexões internas que permitem considerar o histórico de informações em suas previsões. Essa característica as torna particularmente úteis para modelar dependências temporais em aplicações como veículos inteligentes. Entretanto, as RNNs enfrentam desafios em capturar dependências de longo prazo devido à dissipação ou explosão de gradientes durante o treinamento [10].

A Figura 2 apresenta uma ilustração mostrando como as RNNs conectam estados anteriores e presentes. As conexões recorrentes criam uma memória temporal, permitindo a análise de dados sequenciais. Para superar as limitações das RNNs, como o desvanecimento de gradientes, permitindo o aprendizado eficiente de padrões em sequências temporais longas, foram introduzidas as redes *Long Short-Term Memory* (LSTM) e *Gated Recurrent Units* (GRU). As LSTM utilizam células com portas de entrada, esquecimento e saída, que regulam o fluxo de informações dentro da rede. Essas portas permitem que informações relevantes sejam armazenadas por períodos

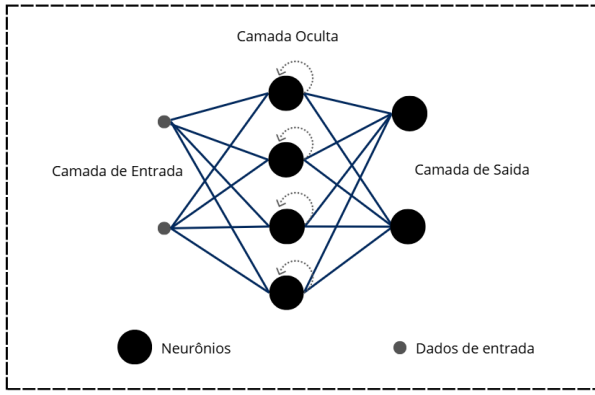


Fig. 2. Estrutura de uma Rede Neural Recorrente.

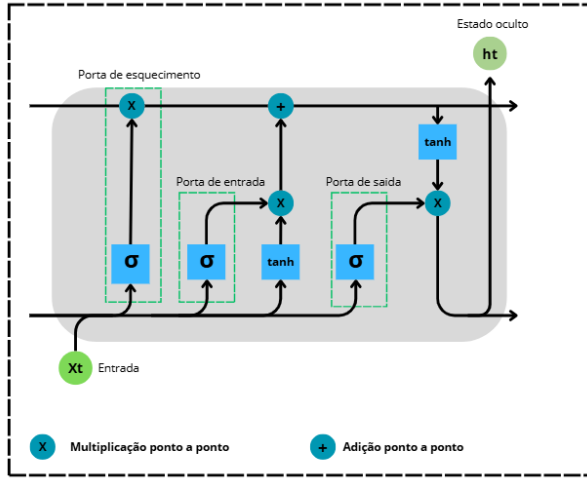


Fig. 3. Diagrama da Rede LSTM.

mais longos, aumentando a capacidade de modelar padrões temporais complexos. A figura 3 mostra a estrutura de uma rede LSTM, conforme [4].

Observa-se na figura 3 que a arquitetura da LSTM é composta por três funções sigmóides (σ) e duas tangentes hiperbólicas ($\tanh$). Realizando operações de soma e multiplicação internas e, tendo como entrada os estados anteriores, obtém-se a predição desejada.

Já as GRUs apresentam uma arquitetura mais simples, combinando as funções das portas de entrada e esquecimento em uma única unidade, reduzindo o número de parâmetros e aumentando a eficiência computacional [4]. Na Figura 4, é apresentado um diagrama da GRU, que ilustra o fluxo de informações dentro da unidade.

Ambas as arquiteturas têm sido amplamente utilizadas em veículos inteligentes [11, 12, 13]. Trabalhos recentes demonstraram como essas redes podem substituir sensores físicos por sensores virtuais, melhorando a precisão e reduzindo os custos dos sistemas de controle [9].

A Tabela I apresenta uma comparação entre as arquiteturas

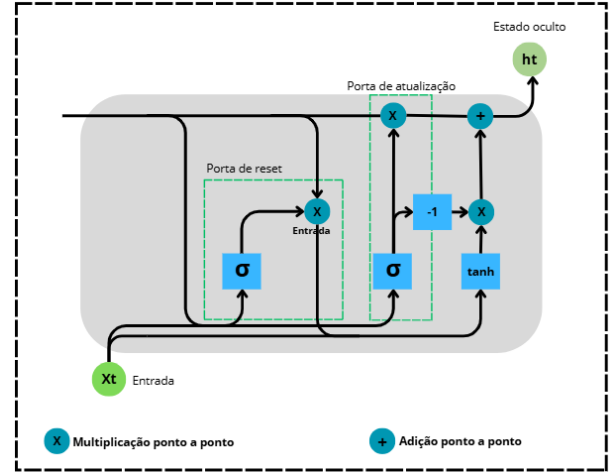


Fig. 4. Estrutura de uma Rede Gated Recurrent Unit.

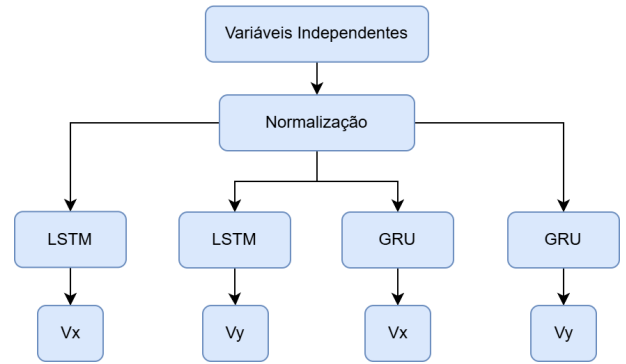


Fig. 5. Estrutura da abordagem de estimativa de estados proposta.

LSTM e GRU, destacando suas principais características, como complexidade, custo computacional e aplicações preferenciais, conforme discutido em [14].

TABLE I
COMPARATIVO ENTRE AS REDES LSTM E GRU.

Aspecto	LSTM	GRU
Complexidade	Alta	Simple
Parâmetros	Mais	Menos
Custo Computacional	Elevado	Eficiente
Memória	Excelente	Boa
Treinamento	Mais lento	Mais rápido
Aplicação	Complexa	Tempo real

III. MATERIAIS E MÉTODOS

Este capítulo apresenta as etapas do desenvolvimento deste trabalho, que englobam a coleta e o pré-processamento dos dados, o desenvolvimento das arquiteturas de redes neurais, e os processos de treinamento e validação. A Figura 5 ilustra a estrutura geral da abordagem proposta para a estimativa de velocidades longitudinal (v_x) e lateral (v_y) em veículos inteligentes.

A. Base de Dados

A base de dados utilizada neste trabalho é resultado de uma parceria com a *University of Jilin*, na China. A plataforma de teste é apresentada na Figura 6. Os módulos de alimentação e aquisição de dados (DAQ) do sistema estão instalados no banco traseiro do veículo. Um sistema GPS/INS é utilizado para fornecer a referência real da velocidade do veículo. A antena do GPS está montada no teto, enquanto a unidade INS está posicionada próxima ao centro de massa do veículo. A plataforma de testes também conta com transdutores de força nos pneus e sensores de um sistema de pneus inteligentes, embora essas medições não sejam utilizadas neste estudo.

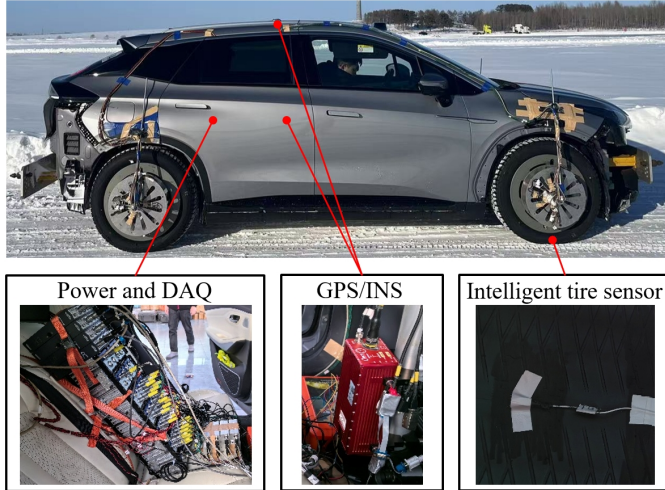


Fig. 6. Plataforma de testes, composta pelo veículo e equipamentos de medição.

Ela é composta por 5 conjuntos, onde cada conjunto representa um teste específico realizado em condições de direção aleatórias (*Random Driving Conditions*). Cada conjunto de dados é composto por sete variáveis, sendo cinco delas consideradas como variáveis de entradas para os modelos:

- Aceleração longitudinal (m/s^2);
- Aceleração lateral (m/s^2);
- Velocidade da roda dianteira (m/s);
- Velocidade da roda traseira (m/s);
- Taxa de guinada (rad/s);

e as outras duas variáveis foram consideradas variáveis de saída, as quais devem ser preditas pelos modelos:

- Velocidade longitudinal (m/s);
- Velocidade lateral (m/s).

O gráfico do comportamento temporal das variáveis do Conjunto 1 é exibido na Figura 7. A frequência de amostragem para todos os 5 conjuntos foi de 200 Hz, diferindo entre si apenas no tempo de aquisição.

B. Pré-Processamento

As variáveis de entrada e saída em um sistema de aquisição de dados frequentemente possuem escalas ou unidades de medida muito diferentes. Essa diversidade nas grandezas

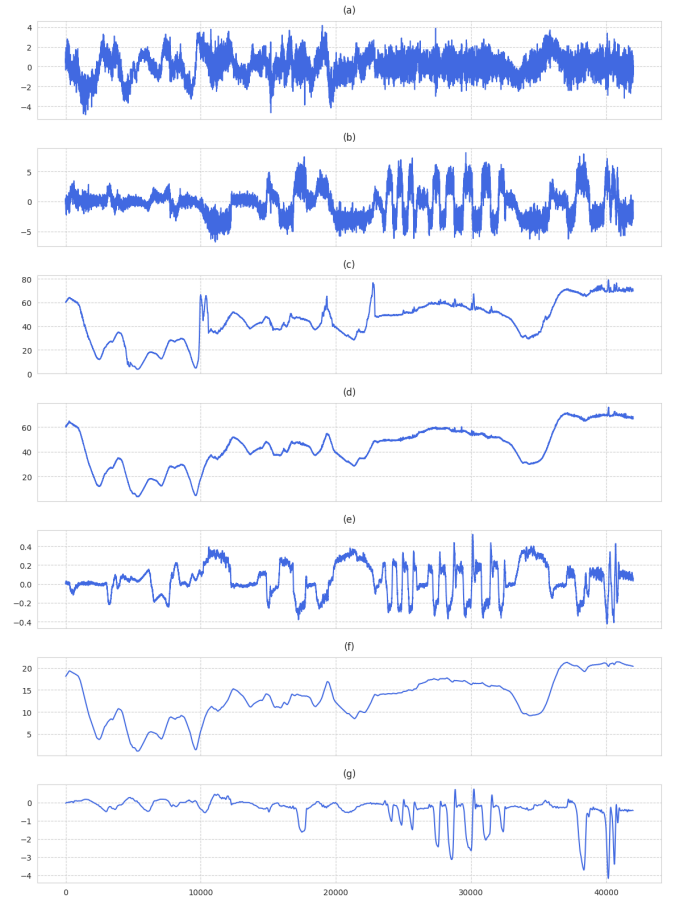


Fig. 7. Representação temporal das variáveis do conjunto de treinamento: Aceleração Longitudinal (a), Aceleração Lateral (b), Velocidade da roda dianteira (c), Velocidade da roda traseira (d), Taxa de guinada (e), Velocidade Longitudinal (f), Velocidade Lateral (g).

físicas pode impactar negativamente o desempenho de redes neurais artificiais, prejudicando o processo de aprendizado e até retardando a convergência durante o treinamento. Por essa razão torna-se essencial realizar o pré-processamento. Uma técnica amplamente utilizada é a normalização de dados, que consiste em ajustar os valores das variáveis para que fiquem em uma escala uniforme, reduzindo assim as desproporções durante o treinamento.

Um pré-processamento eficiente exige uma estratégia bem definida, que passa pela compreensão detalhada dos dados e pela identificação de quais ajustes podem destacar as informações mais relevantes para o processo. Neste estudo, foi adotada a normalização pelo método Máximo Absoluto que visa ajustar os valores das variáveis dividindo-os pelo maior valor absoluto do conjunto de dados, resultando em valores no intervalo de -1 a 1.

A normalização foi aplicada apenas às variáveis de entrada, enquanto as variáveis de saída, que representam os valores preditos pelo modelo, foram mantidas em suas escalas originais para preservar a interpretação direta dos resultados.

Matematicamente, o valor normalizado x' de uma variável

x é calculado por:

$$x' = \frac{x}{|x_{\max}|}, \quad (3)$$

em que x é o valor original da variável e $|x_{\max}|$ é o valor absoluto máximo da variável no conjunto de dados.

C. Modelos de Redes Neurais

Para definir os modelos de redes neurais, foi considerada a capacidade de lidar com dados sequenciais e modelar dependências temporais, característica essencial para a estimativa de estados dinâmicos de veículos. Por essa razão, foram implementadas as duas arquiteturas de redes neurais recorrentes, *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU).

Para maximizar a eficiência e garantir um equilíbrio entre a capacidade de aprendizado, as redes tiveram as mesmas configurações. O número de três camadas, uma camada de entrada com as cinco variáveis de entradas, uma camada oculta e uma camada de saída com um neurônio. Utilizou-se 50 atrasos das variáveis de entrada para a predição das velocidades. Para a camada oculta, foi utilizada a função de ativação ReLU (*Rectified Linear Unit*), definida por:

$$f(x) = \max(0, x). \quad (4)$$

A função ReLU foi escolhida por sua simplicidade computacional e capacidade de evitar problemas de desvanecimento de gradientes, característica comum para redes profundas. Na camada de saída adotou-se a função linear:

$$f(x) = wx + b \quad (5)$$

em que w representa os pesos associados às conexões das camadas anteriores e b é o viés.

O algoritmo de otimização *Adam* foi selecionado por ser amplamente utilizado em problemas de aprendizado profundo devido à sua adaptabilidade e eficiência em ajustar os pesos das redes durante o treinamento. Para evitar *overfitting*, foi aplicada a técnica de *dropout*. Essa técnica foi escolhida porque ajuda a melhorar a generalização dos modelos, reduzindo a dependência de combinações específicas de neurônios durante o treinamento.

D. Métrica de Avaliação

A métrica de avaliação utilizada foi o *Root Mean Square Error* (RMSE), calculada por:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (6)$$

em que y_i representa o valor real, $\hat{y}_i$ é o valor estimado pelo modelo e n é o número total de amostras.

O RMSE é interpretado como uma medida de erro médio, com valores menores indicando maior precisão. Essa métrica é especialmente relevante em problemas de estimação de estados dinâmicos, como os apresentados neste trabalho, pois

fornece uma avaliação quantitativa da precisão das predições em relação aos dados reais.

No contexto de veículos inteligentes, o RMSE é amplamente utilizado para avaliar a precisão de estimativas de velocidades lateral (v_y) e longitudinal (v_x) [4]. Por ser sensível a grandes desvios, ele é uma métrica adequada para identificar discrepâncias em cenários de alta complexidade, como mudanças abruptas de aceleração ou direção.

E. Treinamento e Validação

Os modelos LSTM e GRU foram treinados com um conjunto de dados principal e avaliados em quatro conjuntos de validação. Cada conjunto continha variáveis de entrada, previamente normalizadas pelo método do Máximo Absoluto, e variáveis de saída que permanecem em suas escalas originais. As variáveis de entrada incluíram aceleração longitudinal, aceleração lateral, velocidades das rodas dianteira e traseira, e a taxa de guinada, enquanto as variáveis de saída correspondem às velocidades: longitudinal (v_x) e lateral (v_y).

Em ambos modelos, o treinamento foi realizado em 200 épocas e uma divisão de 20% dos dados para teste durante o treinamento. Durante o treinamento dos modelos, fez-se uso da técnica *Randomized Search* para a busca de hiperparâmetros mais adequados para os modelos. A tabela II mostra o espaço de busca para cada um dos hiperparâmetros selecionados.

TABLE II
ESPAÇO DE BUSCA PARA OS HIPERPARÂMETROS

Hiperparâmetro	Distribuição / Tipo	Valores / Intervalo
batch_size	Lista Discreta	16, 32, 64
Neurônios Ocultos	Distribuição Inteira Uniforme	30 a 99
dropout_rate	Distribuição Contínua Uniforme	0,1 a 0,4
learning_rate	Lista Discreta	0,001, 0,0005, 0,01

O treinamento final do modelo foi feito utilizando a técnica *ModelCheckpoint* para salvar o melhor modelo durante o treinamento. Ademais, aplicou-se a técnica do *EarlyStopping*. Dessa forma, antes de atingir as 200 épocas, caso o treinamento não apresentasse melhorias durante 20 épocas, ele era interrompido.

Após o treinamento, os modelos foram avaliados nos conjuntos de validação. As previsões dos modelos foram comparadas aos valores reais, utilizando o RMSE. Essa abordagem permitiu analisar a precisão dos modelos e sua capacidade de generalização ao trabalhar com dados de diferentes conjuntos.

IV. RESULTADOS E DISCUSSÕES

A tabela III mostra os melhores hiperparâmetros encontrados para cada um dos modelos LSTM e GRU treinados para a predição das velocidades v_x e v_y . É possível observar que, dentro o espaço de busca do *learning rate*, os modelos GRU e LSTM obtiveram a mesma taxa para suas respectivas predições de v_y e v_x . Para esse caso, isso demonstra que o hiperparâmetro *learning rate* está mais associado ao modelo de treinamento do que à variável de saída a ser predita no caso em questão. Além disso, observa-se também que os modelos LSTM precisaram de mais camadas ocultas do que o GRU

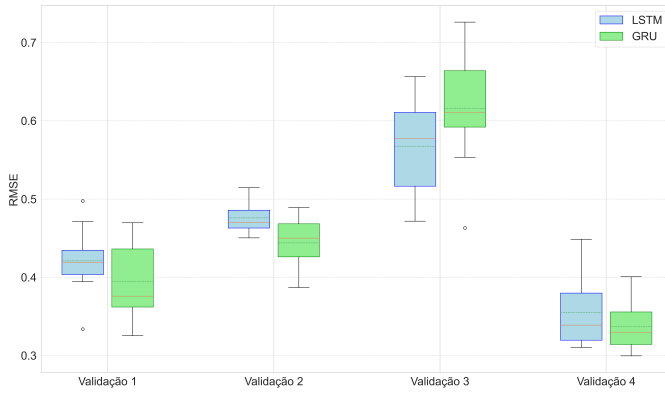


Fig. 8. Comparação do RMSE para v_x nos conjuntos de validação em 10 rodadas de treinamento.



Fig. 9. Comparação do RMSE para v_y nos conjuntos de validação em 10 rodadas de treinamento.

para obter melhores resultados. Os valores mais adequados encontrados para o *dropout rate* não apresentam variações significativas entre os quatro modelos, todos se apresentando entre o intervalo de 10,5% ao máximo de 14,6%.

TABLE III
MELHORES HIPERPARÂMETROS ENCONTRADOS PARA CADA MODELO

Hiperparâmetro	GRU v_x	GRU v_y	LSTM v_x	LSTM v_y
batch_size	16	32	64	16
dropout_rate	0,124	0,105	0,146	0,109
Neurônios Ocultos	41	44	75	66
learning_rate	0,001	0,001	0,0005	0,0005

As Figuras 8 e 9 apresentam a comparação do RMSE para v_x e v_y nos conjuntos de validação, utilizando as arquiteturas LSTM e GRU. Cada modelo foi treinado 10 vezes, resultando na obtenção de dez valores distintos do RMSE. Estes valores foram utilizados na construção dos *boxplot*.

A Figura 8 apresenta o RMSE dos conjuntos de validações para a variável v_x , permitindo uma análise detalhada do desempenho das arquiteturas LSTM e GRU. Observa-se que, para todos os quatro conjuntos de validação, os modelos apresentam valores de RMSE próximos entre si para a predição da velocidade v_x . O conjunto de validação 4 é o que apresenta

os maiores valores de RMSE, enquanto que o conjunto de validação 2 apresenta os menores valores. Isso demonstra a eficiência de ambos os modelos, porém, enfatiza ainda mais o GRU, tendo em vista que o modelo necessitou de menos neurônios nas camadas ocultas.

Já na Figura 9, é mostrado o RMSE das validações para a variável v_y usando as arquiteturas LSTM e GRU. Ambos os modelos mantiveram desempenhos similares para os conjuntos de validações. No conjunto de validação 1, LSTM apresentou uma variância maior nos resultados. O mesmo ocorreu com o GRU, porém, para o conjunto de validação 4. O conjunto de validação 3 se mostrou o menos complexo para a modelagem do comportamento de v_y , refletindo nos menores valores de RMSE mostrados pelo boxplot.

As Figuras 10 a 17 detalham as predições para os conjuntos de validação, comparando os valores reais e preditos.

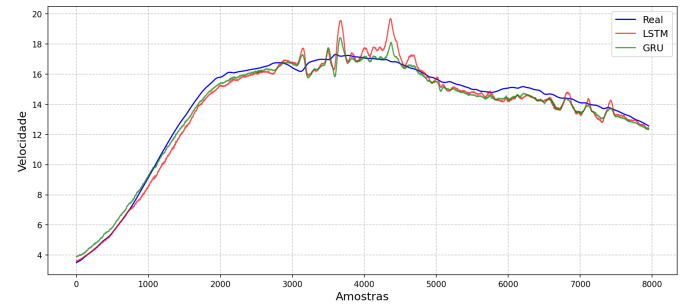


Fig. 10. Comparativo entre Valores Preditos e Valores Reais de v_x do Conjunto de Validação 1

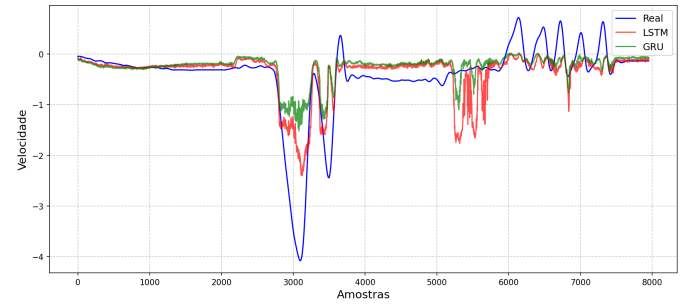


Fig. 11. Comparativo entre Valores Preditos e Valores Reais de v_y do Conjunto de Validação 1

A partir das Figuras 10, 12, 14 e 16 nota-se que, para a variável v_x , as duas arquiteturas acompanham bem as flutuações dos dados reais.

Assim como para a variável v_x , as predições de v_y também foram similares em ambos modelos. No entanto, conforme pode ser observados nas Figuras 11, 13, 15 e 17, a predição dessa variável é mais desafiadora, pois há alterações bruscas e os modelos não conseguem acompanhar bem essas maiores alterações, principalmente ao observar o resultado no conjunto de validação 4. Estes resultados reforçam a eficiência de ambos modelos para predição da velocidade longitudinal, mas mostra

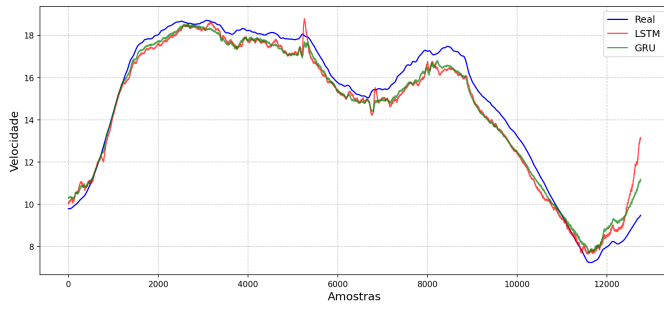


Fig. 12. Comparativo entre Valores Preditos e Valores Reais de v_x do Conjunto de Validação 2

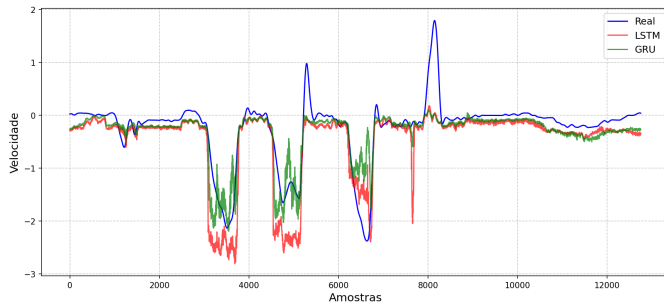


Fig. 13. Comparativo entre Valores Preditos e Valores Reais de v_y do Conjunto de Validação 2

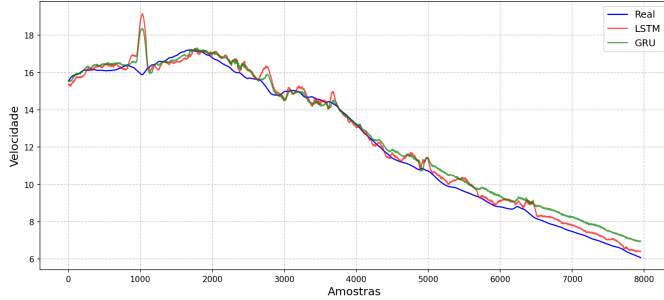


Fig. 14. Comparativo entre Valores Preditos e Valores Reais de v_x do Conjunto de Validação 3

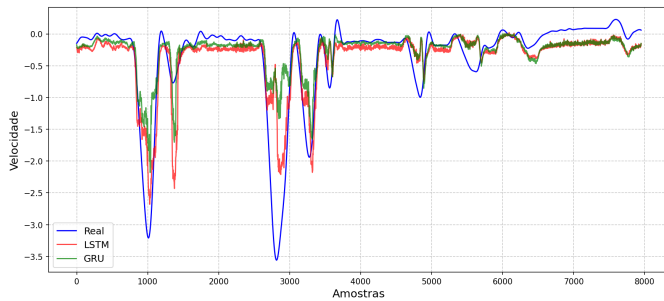


Fig. 15. Comparativo entre Valores Preditos e Valores Reais de v_y do Conjunto de Validação 3

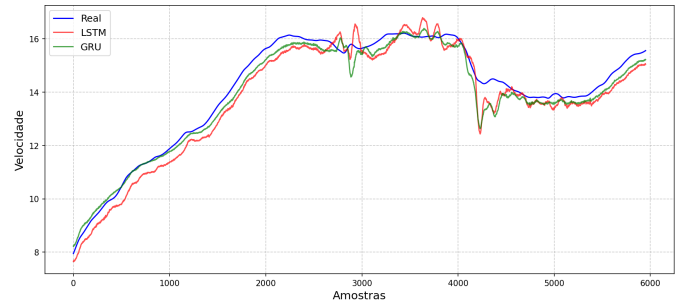


Fig. 16. Comparativo entre Valores Preditos e Valores Reais de v_x do Conjunto de Validação 4

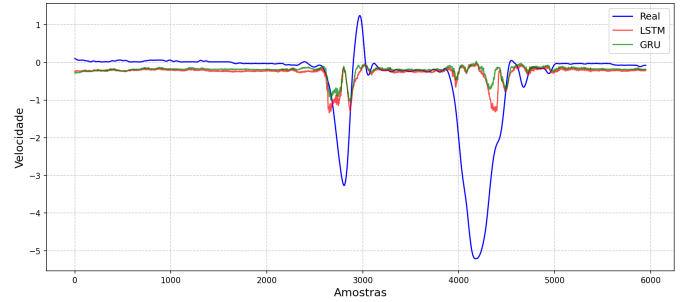


Fig. 17. Comparativo entre Valores Preditos e Valores Reais de v_y do Conjunto de Validação 4

também que maiores estudos devem ser feitos para a predição de velocidade lateral.

V. CONCLUSÃO

Este trabalho explorou o uso de aprendizado de máquinas, especificamente as arquiteturas de redes neurais LSTM e GRU, para estimar estados dinâmicos em veículos inteligentes, foram eles: as velocidades longitudinal (v_x) e lateral (v_y). Constatou-se que ambas as arquiteturas são capazes de estimar a velocidade longitudinal de forma satisfatória, com baixos valores de RMSE em validação, apresentando bons desempenhos na modelagem de dependências temporais complexas de v_x , evidenciando suas capacidades em capturar padrões intrincados nos dados. Em relação à predição de v_y , os modelos conseguiram estimá-la em situações de baixa excitação lateral, no entanto, em mudanças bruscas da velocidade lateral, os modelos não conseguiram acompanhar bem a dinâmica, principalmente em valores extremos.

A partir dos resultados, tanto a rede LSTM e GRU apresentaram desempenhos estatisticamente similares. Entretanto, pode-se destacar o modelo GRU, que possui uma arquitetura mais simples, dispondo de menos neurônios em suas camadas ocultas, refletindo em um modelo com menor custo computacional se comparado ao LSTM.

Fica claro, portanto, que o potencial das redes neurais na substituição de sensores físicos por sensores virtuais oferece uma solução eficiente e de menor custo para veículos inteligentes. Futuras pesquisas podem expandir este trabalho,

explorando outras arquiteturas de redes neurais e avaliando o impacto de diferentes técnicas de pré-processamento e treinamento em condições mais diversificadas, principalmente para estimação da dinâmica lateral.

VI. AGRADECIMENTOS

Agradecimento à FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais), ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e à CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelos recursos financeiros destinados à execução do projeto.

REFERENCES

- [1] T. Taulli, *Introdução à Inteligência Artificial: Uma Abordagem Não Técnica*. São Paulo: Novatec Editora, 2021.
- [2] Y. Wang, H. Wei, L. Yang, B. Hu, and C. Lv, "A review of dynamic state estimation for the neighborhood system of connected vehicles," *SAE International Journal of Vehicle Dynamics, Stability, and NVH*, 2023.
- [3] K. Singh, M. Arat, and S. Taheri, "Literature review and fundamental approaches for vehicle and tire state estimation," *Vehicle System Dynamics*, vol. 57, pp. 1–23, 11 2018.
- [4] D. Kong, W. Wen, R. Zhao, Z. Lv, K. Liu, Y. Liu, and Z. Gao, "Vehicle lateral velocity estimation based on long short-term memory network," *World Electric Vehicle Journal*, vol. 13, no. 1, pp. 1–10, 2022.
- [5] X.-B. Jin, R. J. Robert Jeremiah, T.-L. Su, Y.-T. Bai, and J.-L. Kong, "The new trend of state estimation: From model-driven to hybrid-driven methods," *Sensors*, vol. 21, no. 6, 2021.
- [6] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2021.
- [7] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed., ser. Mechanical Engineering Series. Springer, 2012.
- [8] W. Jiang, C. Xu, and Y. Li, "Comparison of kinematic and dynamic models for vehicle motion estimation," *IEEE Transactions on Vehicular Technology*, vol. 68, 2019.
- [9] G. Napolitano Dell'Annunziata, V. M. Arricale, F. Farroni, A. Genovese, N. Pasquino, and G. Tranquillo, "Estimation of vehicle longitudinal velocity with artificial neural network," *Sensors*, vol. 22, no. 23, p. 9516, 2022.
- [10] M. A. de Oliveira and F. A. de Pulpa Melo Junior, *Back-propagation e Redes Neurais: MLP, RPROP, QPROP, LMBP, Delta-Bar-Delta, MATLAB e Python*. Rio de Janeiro: Editora Ciência Moderna, 2020, vol. 1.
- [11] M. D. Lio, M. Piccinini, and F. Biral, "Robust and sample-efficient estimation of vehicle lateral velocity using neural networks with explainable structure informed by kinematic principles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 13 670–13 684, 2023.
- [12] R. Song, Y. Fang, and H. Huang, "Reliable estimation of automotive states based on optimized neural networks and moving horizon estimator," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 6, pp. 3238–3249, 2023.
- [13] T. Gräber, S. Lupberger, M. Unterreiner, and D. Schramm, "A hybrid approach to side-slip angle estimation with recurrent neural networks and kinematic vehicle models," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 1, pp. 39–47, 2019.
- [14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.