

Automated Vin Identification Approach With Computer Vision For Vehicle Recognition

Lucas Buligon Antunes¹, Guilherme Loan Schneider¹, Igor Wiese²,
Thiago França Naves¹, Anderson Soares³

¹Departamento de Ciência de Computação – UTFPR Campus Santa Helena
Santa Helena, Paraná, Brasil

²Departamento de Ciência de Computação – UTFPR Campus Campo Mourão
Paraná, Brasil

³Departamento de Ciência da Computação – UFG - INF Campus Samambaia
Goiás, Brasil

{lucasbuligon, guilhermelloan}@alunos.utfpr.edu.br, naves@utfpr.edu.br,
igor@professores.utfpr.edu.br, andersonsoares@ufg.br

Abstract. *Driven by the need to handle vehicle fraud, which exceeded R\$504 million in the first half of 2024 in Brazil, this work presents an automated computer vision system for chassis number detection and recognition. The proposed solution is structured as a five-step pipeline, encompassing the localization of the region of interest using Oriented Bounding Boxes (OBB) and its orientation correction, Optical Character Recognition (OCR), and a final refinement stage through business rule-based post-processing. The main contribution lies in demonstrating that training enrichment with synthetic data is crucial for ensuring system robustness in real-world scenarios, allowing the pipeline to achieve 96% accuracy in full chassis number reading. Future improvements will focus on optimizing the accuracy for perfect reading and exploring end-to-end architectures to simplify the solution and assess its real-time computational performance.*

Resumo. *Impulsionado pela necessidade de combater fraudes veiculares, que ultrapassaram R\$ 504 milhões no primeiro semestre de 2024 no Brasil, este trabalho apresenta um sistema automatizado de visão computacional para a detecção e o reconhecimento de números de chassi. A solução é estruturada em uma pipeline de cinco etapas, que abrange desde a localização da região de interesse com Caixas Delimitadoras Orientadas (OBB) e a correção de sua orientação, até o reconhecimento óptico de caracteres (OCR) e um refinamento final via pós-processamento com regras de negócios. A principal contribuição reside na demonstração de que o enriquecimento do treinamento com dados sintéticos é crucial para a robustez do sistema em cenários reais, permitindo que a pipeline alcance 96% de acurácia na leitura completa do chassi. Futuras melhorias se concentrarão na otimização da acurácia para uma leitura perfeita e na exploração de arquiteturas end-to-end para simplificar a solução e analisar seu desempenho computacional em tempo real.*

1. Introdução

A leitura automatizada de informações textuais em ambientes não controlados permanece como um desafio central na área de visão computacional. Um exemplo crítico dessa di-

ficuldade é o reconhecimento automático do Número de Identificação Veicular (VIN), também conhecido como número do chassi. O VIN é essencial para a rastreabilidade, autenticação e prevenção de fraudes em veículos, sendo utilizado amplamente nos setores logístico, automobilístico e de segurança [American Journal of Transportation 2024, Lubiato 2015]. A relevância dessa questão é corroborada pelos dados do mercado segurador brasileiro, onde, apenas no primeiro semestre de 2024, o valor das fraudes comprovadas somou R\$ 504 milhões, destacando a necessidade de sistemas mais eficientes para a detecção dessas ocorrências [Confederação Nacional das Seguradoras 2024].

Neste trabalho, o foco consiste na leitura de VINs gravados diretamente na estrutura metálica do veículo, um tipo de gravação frequentemente encontrado em inspeções veiculares, especialmente em situações de verificação física do automóvel. Embora a gravação em metal não seja exclusiva do Brasil, estudos como o de [Ramshankar and Deivanathan 2018] utilizam conjuntos de dados contendo VINs impressos em etiquetas adesivas, com condições visuais mais favoráveis, como fundo plano, boa iluminação e tipografia uniforme. Em contraste, o presente trabalho enfrenta o desafio de imagens reais capturadas em condições adversas, como iluminação desigual, ruídos, ângulos variados e ampla variabilidade tipográfica, especialmente comuns no contexto brasileiro.

Para lidar com essas adversidades, propõe-se uma *pipeline* com cinco etapas sequenciais. O processo inicia com a detecção da região do chassi utilizando Caixas Delimitadoras Orientadas (OBBs) para lidar com a variação de ângulos, seguido por uma etapa de correção automatizada da orientação do texto. Subsequentemente, realiza-se a detecção dos caracteres individuais. Após isso, a classificação dos caracteres, um processo otimizado pela aplicação de *padding* nos recortes. Por fim, um módulo de pós-processamento integra regras de negócios, baseadas na norma NBR 6066, para refinar e validar o resultado final, demonstrando a eficácia da combinação de aprendizado profundo com conhecimento de domínio para resolver um problema industrial complexo.

O desenvolvimento desta solução atende a uma demanda específica de uma empresa parceira do Laboratório de Aprendizado de Máquina Aplicado à Indústria (LAMIA), sendo o trabalho aplicado na criação de uma solução para seu contexto operacional. O sistema foi treinado e validado utilizando um conjunto de imagens reais cedido pela parceira, refletindo os desafios de um ambiente de inspeção veicular.

A principal contribuição deste trabalho é a demonstração quantitativa do impacto de dados sintéticos na robustez de sistemas de Reconhecimento Óptico de Caracteres (OCR) para aplicações industriais. Ao gerar um conjunto de 5.000 imagens sintéticas e integrá-lo a dados reais de chassis, a *pipeline* proposta elevou o desempenho da detecção de caracteres a um patamar de 99,72% de acurácia e 0,9993 de sensibilidade. Essa estratégia de enriquecimento de dados é o pilar da *pipeline* completa desenvolvida neste estudo, que é complementada por otimizações, como a aplicação de *padding* para aprimorar a classificação e a integração de regras de negócios no pós-processamento, elementos que se mostraram essenciais para a alta performance final do sistema.

2. Trabalhos Relacionados

As primeiras abordagens na literatura para automatizar a leitura do VIN combinaram técnicas clássicas de processamento de imagem com redes neurais artificiais iniciais. O

trabalho de [Shah et al. 2009] apresentou um algoritmo para identificação de VINs de 5 dígitos gravados (em relevo) utilizando Redes Neurais Artificiais (RNA). O desafio residia na dificuldade de decifrar os caracteres nessas condições. Para isso, foram utilizadas mais de mil imagens capturadas por uma câmera, que passavam por um pré-processamento de detecção de bordas (Sobel), segmentação e normalização. A classificação era realizada por uma RNA de três camadas, alcançando Taxa de Identificação Correta (CIR) de 95,49% com zero erros de identificação para os caracteres numéricos testados [Shah et al. 2009].

Por sua vez, [Souza et al. 2014] propuseram uma ferramenta de baixo custo para validação de números de chassi em linhas de produção. A solução utilizava uma webcam e um software desenvolvido em C# com as bibliotecas Emgu CV e Tesseract-OCR. A *pipeline* incluía a identificação da Região de Interesse (ROI) por meio de operações morfológicas (erosão/dilatação) e detecção de bordas com Canny. Na primeira etapa, a taxa de sucesso foi de 80% na resolução de 480p e caiu para 60% em 1280p, limitada principalmente pela iluminação não controlada. Já o Tesseract OCR obteve 100% de acerto nas ROIs que foram detectadas com sucesso.

Já [Patil and Dhanvijay 2015] focaram na robustez para aplicações em tempo real sob condições de iluminação desfavoráveis. O sistema proposto, desenvolvido em JAVA, também utilizava uma *pipeline* tradicional de pré-processamento para extrair e segmentar os caracteres, seguido por um sistema OCR para identificá-los com base em um modelo pré-treinado. Utilizando imagens de câmeras de celular, os autores relataram uma precisão de até 99,99% em seus testes. A alta acurácia é justificada pelo fato de o sistema funcionar apenas para veículos cujos números já estão armazenados em um banco de dados e por obter os resultados corretos ao fazer a correspondência com esse banco de dados de imagens original.

O trabalho de [Ramshankar and Deivanathan 2018], por exemplo, descreveu o desenvolvimento de um sistema de visão para inspecionar adesivos de VIN em portas de carro, combinando uma câmera móvel controlada por Arduino com um OCR baseado em correspondência de modelo. Embora utilizasse uma técnica clássica de reconhecimento, a automação do sistema de captura representou um avanço, alcançando 100% de precisão para um VIN de 17 caracteres em um tempo de processamento de 3 a 4 segundos [Ramshankar and Deivanathan 2018].

[Yang and Feng 2020] propuseram um sistema de reconhecimento de números de motor em um único estágio, utilizando Redes Neurais Convolucionais (CNN) e aprendizado por transferência. O método elimina a necessidade de pré-processamento e segmentação explícita de caracteres e alcançou uma alta precisão de 99,48% em seus testes, com um tempo de execução de aproximadamente 230 ms por imagem [Yang and Feng 2020]. Como o estudo trabalha com números de motor, o conjunto de caracteres reconhecíveis era limitado a 14 classes específicas e os autores deixaram de trabalhar a generalização do sistema para uma variedade maior de motores e a sua aplicação em tempo real.

No estudo de [Prakash et al. 2023], os autores desenvolveram um método focado em melhorar a precisão sob diferentes condições de iluminação, elevando a taxa de acerto de 64,28% para cerca de 85-90% através de uma técnica de aprimoramento manual de

imagem. A abordagem é voltada para o reconhecimento de números de motores. Além disso, a validação do sistema se deu em um cenário restrito de 28 fotos capturadas em um ambiente controlado por uma grade, deixando em segundo plano um teste em condições mais variadas e sem intervenção manual.

O atual trabalho, inspirado por essas contribuições, propõe uma *pipeline* que utiliza conjuntos de imagens sintéticas e reais, com o objetivo de treinar os modelos de correção de orientação, reconhecimento da região de interesse e reconhecimento de caracteres, a fim de obter resultados superiores de precisão quando comparados com os trabalhos anteriormente citados que utilizaram apenas dados reais. Além disso, é importante notar a utilização de OBBs, que são empregadas no processo de correção de orientação das imagens, e a aplicação de *padding*, a fim de melhorar o desempenho do algoritmo.

3. Metodologia

A Figura 1 ilustra a *pipeline* proposta neste trabalho para a leitura automática do VIN, abaixo são descritos cada um de seus processos. Ela começa com a leitura do conjunto de imagens, possuindo 4 etapas para o reconhecimento dos caracteres do chassi e uma etapa subsequente de validação a partir de regras de negócios, até a sua representação final na etapa 7. Por força de um Acordo de Confidencialidade (NDA), aspectos proprietários da implementação, incluindo detalhes sobre as ferramentas utilizadas, são omitidos neste artigo, que foca nos resultados e na metodologia de aprendizado de máquina aplicada.

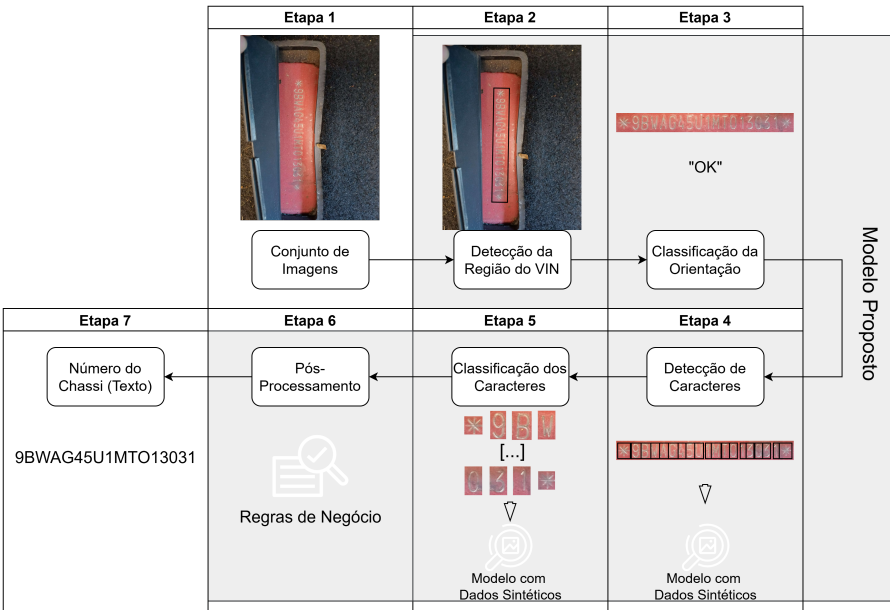


Figura 1. Etapas do processamento dos dados no atual trabalho

3.1. Detecção do Chassi

A etapa inicial do processo consiste na detecção do VIN. Para essa finalidade, emprega-se um modelo de detecção de objetos baseado em OBBs. A localização do chassi em veículos frequentemente apresenta desafios de acessibilidade, resultando em variações significativas nos ângulos de captura das imagens. Consequentemente, a correção da orientação da região detectada torna-se uma etapa crucial.

A utilização de OBBs permite não apenas o recorte preciso da região de interesse contendo o chassi, mas também a retificação do ângulo de inclinação. Esse procedimento assegura que o texto presente na gravação do chassi seja apresentado de forma alinhada e ortogonal, facilitando análises subsequentes e o processamento automatizado, ao concentrar-se exclusivamente na área relevante da imagem.

O treinamento do modelo de detecção de chassi foi conduzido utilizando um conjunto de dados abrangente, composto por 7.588 imagens positivas (contendo instâncias de chassi) e 1.365 imagens negativas (sem a presença de chassi). Essas imagens são particularmente relevantes por terem sido fornecidas diretamente por uma empresa parceira. Esse conjunto de dados foi dividido nas seguintes proporções: 65% para treino, 15% para validação e 20% para teste. Um aspecto crucial do processo foi a inclusão das imagens negativas. Treinar o modelo apenas com imagens positivas resultava em uma alta taxa de falsos positivos, e a adição estratégica dessas amostras negativas foi fundamental para mitigar esse problema.

3.2. Correção da Orientação

Apesar da capacidade do modelo OBB em delimitar a região do chassi, uma limitação intrínseca impedia o retorno da orientação correta do texto contido. Frequentemente, ele era detectado em posições invertidas, comprometendo a legibilidade e a subsequente análise automatizada. Para mitigar essa limitação, foi treinada uma rede de classificação para a correção da orientação.

Esta CNN avalia a ROI detectada pelo modelo OBB categorizando-a em uma de duas classes: “orientação correta” ou “invertida”. Caso a ROI seja classificada como “invertida”, é aplicada uma rotação de 180° para corrigir sua orientação. Essa abordagem garante que o texto do chassi seja sempre apresentado de forma padronizada, facilitando o processamento subsequente e elevando a eficácia do sistema como um todo.

Para o treinamento do classificador responsável pela correção da orientação do texto do chassi, foram coletadas e anotadas, juntamente à empresa parceira, 3405 novas imagens de chassis já recortadas e com a orientação textual correta. Para criar a classe correspondente a “orientação invertida”, cada uma dessas imagens foi rotacionada 180°. Essa abordagem garantiu um conjunto de dados balanceado para ambas as classes (“orientação correta” e “invertida”). O conjunto de dados combinado foi subsequentemente dividido em subconjuntos para treino, validação e teste, seguindo as proporções de 65% para treino, 15% para validação e 20% para teste.

3.3. Reconhecimento Óptico de Caracteres (OCR)

A tarefa de OCR foi estruturada em duas etapas sequenciais: primeiramente, a detecção dos caracteres individuais, seguida pela classificação de cada caractere recortado. Para a fase de detecção de caracteres, quatro modelos distintos foram treinados e avaliados, utilizando diferentes configurações de conjuntos de dados:

Modelos Baseados em Dados Reais

O primeiro modelo foi treinado exclusivamente com imagens reais fornecidas pela empresa parceira. Para isso, empregou-se um novo conjunto de 1380 imagens de chassis,

das quais a área do texto foi previamente recortada, contendo as caixas delimitadoras dos caracteres individuais. Essas imagens foram divididas em subconjuntos para treino, validação e teste, seguindo as proporções de 65% para treino, 15% para validação e 20% para teste, respectivamente.

No segundo modelo, além das imagens reais utilizadas anteriormente, incorporou-se um volume adicional de 658 imagens reais provenientes da internet. Essas imagens foram adicionadas ao split de treino, seguindo a mesma metodologia de preparação e distribuição dos dados empregada no modelo anterior.

Modelos com Dados Sintéticos

O terceiro modelo expandiu o conjunto de dados de treinamento com a inclusão de 5000 imagens sintéticas. Essas imagens foram modeladas e renderizadas utilizando o software Blender, sendo integradas ao split de treino. É possível visualizar o exemplo de uma imagem sintética gerada para o treinamento e sua máscara na Figura 2, com ela é possível gerar a anotação da imagem automaticamente.

A adição de dados sintéticos visa aumentar a robustez e a capacidade de generalização do modelo, expondo-o a uma maior variabilidade de estilos e condições de iluminação. Este modelo, no conjunto de testes composto por 5347 instâncias, obteve 99% de precisão, 1 de sensibilidade e 0.99 de F1-score, demonstrando excelente desempenho.



Figura 2. Exemplo de imagem sintética de chassi.

Adicionalmente, um quarto modelo foi treinado apenas com dados sintéticos para fins comparativos. Embora este modelo tenha apresentado a menor *loss* (diferença entre as saídas previstas e esperadas do modelo [Terven et al. 2025]) observada durante o treinamento, conforme ilustrado na Figura 3, seu desempenho em dados reais foi insatisfatório. Esse comportamento é atribuído à mudança de domínio entre os dados de treinamento (sintéticos) e os dados de teste (reais), um fenômeno comum que dificulta a generalização do modelo para dados de distribuição diferente.

Essa abordagem incremental na composição dos conjuntos de dados de treinamento, explorando diferentes volumes e tipos de imagens (reais e sintéticas), permitiu uma análise comparativa aprofundada da influência desses fatores na performance da detecção de caracteres para o reconhecimento óptico de texto do chassi.

A etapa de classificação individual de caracteres utilizou o mesmo conjunto de dados empregado para a detecção de caracteres (imagens reais, imagens da internet e imagens sintéticas). Para tal, as detecções de cada caractere foram recortadas e organizadas em diretórios rotulados com suas respectivas classes. Nesta fase, três modelos de classificação distintos foram treinados, visando otimizar a acurácia com base na posição do caractere no chassi.

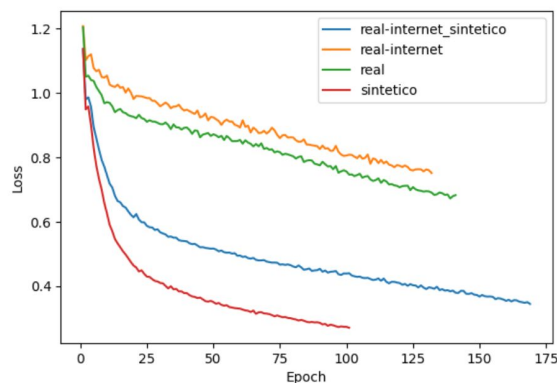


Figura 3. Loss dos modelos durante o treinamento

O primeiro modelo foi treinado especificamente para a classificação de números. O segundo modelo, por sua vez, focou na classificação de caracteres alfanuméricos (letras e números). Por fim, o terceiro modelo abordou a distinção entre caracteres alfanuméricos e caracteres especiais. A importância deste último reside no fato de que os caracteres especiais podem ocorrer no início e no final do número de chassi e sua simbologia pode variar entre diferentes montadoras. Sabendo da posição do caractere especial, é possível inferir corretamente a posição do restante dos caracteres.

Com o objetivo de aprimorar ainda mais o desempenho desses classificadores, foi conduzida uma análise complementar sobre o impacto do redimensionamento da caixa delimitadora dos caracteres antes do recorte para classificação. Foram avaliadas três estratégias: o recorte direto da imagem do caractere, o recorte após um aumento de 10% apenas na altura da caixa delimitadora, e o recorte após um aumento de 10% tanto na largura quanto na altura da caixa delimitadora, denominado “padding”.

3.4. Pós-processamento

Após a detecção e classificação individual dos caracteres, as informações são concatenadas para formar um resultado preliminar do número de chassi. Subsequentemente, esse resultado passa por uma etapa de pós-processamento, que visa refinar e validar o resultado final. Os tratamentos aplicados ao número de chassi são embasados em uma análise exploratória detalhada, realizada conforme as diretrizes da NBR 6066 [Associação Brasileira de Normas Técnicas 2001] e os dados veiculares disponíveis. Essa análise permite inferir informações cruciais sobre os nove primeiros dígitos do chassi a partir de outros dados do veículo, como montadora, modelo e motorização. A capacidade de prever ou confirmar partes dos 3 primeiros dígitos (o *World Manufacturer Identifier* - WMI) com base nessas informações auxilia significativamente na identificação precisa do início do chassi, o que, por sua vez, reduz a ocorrência de erros na transcrição completa. A Figura 4 exemplifica o processo de descarte de um caractere especial, fundamentado na validação pelo WMI. Em testes posteriores, a aplicação desse processo de pós-processamento resultou em um aumento de 5% na acurácia geral da leitura do chassi, demonstrando a eficácia das regras e validações para aprimorar a confiabilidade do sistema.



Figura 4. Detecção de caractere sendo descartada com base no WMI

4. Resultados e Discussão

O sistema de leitura de chassi foi desenvolvido como uma *pipeline* de múltiplos estágios, onde o desempenho de cada componente individual é fundamental para o sucesso do resultado final. A seguir, o desempenho de cada etapa é detalhado, culminando na avaliação do sistema integrado. A detecção do chassi (etapa 2), representada na Figura 1, demonstrou alta eficácia. Avaliado em um conjunto de 1791 amostras, o modelo alcançou uma acurácia de 98,18%. Complementarmente, a precisão de 98,40% minimiza a incidência de falsos positivos. A sensibilidade de 0,9977 é particularmente significativa, atestando que o modelo identificou a vasta maioria dos chassis existentes no conjunto de teste. O F-score de 0,9908, como média harmônica entre precisão e sensibilidade, confirma o equilíbrio do modelo entre evitar falsas detecções e recuperar instâncias reais.

Após a localização do chassi, a etapa seguinte é a classificação da sua orientação (etapa 3), representada na Figura 1. O modelo classificador responsável por essa tarefa alcançou uma acurácia de 99%, com precisão, sensibilidade e F1-score também em 0,99. Esses valores uniformemente elevados garantem que o texto do chassi seja padronizado com alta confiabilidade, minimizando erros de orientação que poderiam comprometer as etapas subsequentes. É importante ressaltar que os altos valores de métricas alcançados por este classificador são explicados pela relativa simplicidade do problema a ser resolvido. A tarefa de identificar se uma imagem está na orientação correta ou rotacionada em 180° é inerentemente mais direta em comparação com problemas de detecção ou classificação multiclasse mais complexos, o que justifica os resultados obtidos.

Com o chassi localizado e corretamente orientado, a *pipeline* avança para a detecção dos caracteres individuais (etapa 4), representada na Figura 1. Dentre os modelos treinados, a abordagem que combinou dados reais, imagens da internet e dados sintéticos apresentou o melhor desempenho. Avaliado em 5.347 instâncias de imagens reais, este modelo alcançou 99% de acurácia, 0,999 de sensibilidade e 0,99 de F1-score. A inclusão de dados sintéticos no treinamento foi, portanto, uma estratégia decisiva para aumentar a robustez e a generalização do modelo em cenários reais, como demonstra a Tabela 1.

Uma vez detectados, os caracteres são enviados para a classificação (etapa 5), representados na Figura 1. Para otimizar essa etapa, foi adotada uma abordagem estratégica com múltiplos classificadores específicos, alinhados à estrutura do chassi. Essa especialização permitiu alcançar 97% de acurácia para caracteres numéricos, 98% para alfanuméricos e 99% para a distinção entre alfanuméricos e especiais. Um fator crucial

Tabela 1. Desempenho dos Modelos de Detecção de Caracteres por Conjunto de Dados

Conjunto	Acurácia	Sensibilidade	F1-Score
Real	99,40%	0,9991	0,9970
Real + Internet	94,89%	0,9517	0,9737
Real + Internet + Sintético	99,72%	0,9993	0,9986

para atingir essa performance foi a expansão das caixas delimitadoras (*padding*) em 10% antes do recorte, o que forneceu contexto visual adicional ao classificador, como ilustrado na Tabela 2. A Tabela 3 evidencia que a combinação de todos os conjuntos de dados com a técnica de *padding* foi a que obteve os melhores resultados.

Tabela 2. Comparativo de acurácia na classificação de caracteres alfanuméricos com diferentes estratégias de *padding*.

Estratégia de <i>Padding</i>	Acurácia
Sem <i>padding</i>	93,28%
<i>Padding</i> na altura	97,00%
<i>Padding</i> na altura e largura	98,00%

Tabela 3. Desempenho dos Modelos de Classificação de Caracteres por Conjunto de Dados

Conjunto	Acurácia	Precisão
Internet	69.49%	72.68%
Sintético	46.84%	56.91%
Internet + Sintético	78.62%	80.70%
Internet + Sintético + Real	93.28%	93.31%
Internet + Sintético + Real (com <i>padding</i>)	98%	98%

Após o treinamento e a otimização individual de cada modelo, eles foram integrados na avaliação da *pipeline* completa. Em um conjunto de teste com 500 imagens reais e inéditas (não integradas nos testes anteriores), o sistema alcançou uma acurácia de 79,05% para a leitura perfeita dos 17 caracteres do chassi. Embora a precisão total seja uma métrica importante, uma análise mais granular revela a robustez prática do sistema: a acurácia sobe para 92,76% ao permitir um único erro e para 96% com tolerância de até dois erros.

Para aprofundar a análise de desempenho, a acurácia foi calculada por posição de caractere, como detalhado na Tabela 4. Os dados mostram que um pós-processamento aplicado aos nove primeiros caracteres contribuiu para manter a acurácia média dessa seção 4,2% superior à do restante do chassi. A análise da matriz de confusão na Figura 5 revela que a maior parte dos erros de classificação ocorre entre caracteres visualmente semelhantes, como "7" e "1", principalmente na seção numérica final. É notável que, em

um universo de 8.500 caracteres (500 imagens x 17 caracteres), o número absoluto de confusões demonstrou ser baixo.

Tabela 4. Acurácia por Posição do Caractere no Chassi

Posição	Acurácia (%)
1	99.05
2	98.86
3	99.05
4	99.24
5	97.33
6	99.05
7	97.33
8	98.10
9	97.52
10	94.86
11	95.62
12	96.19
13	96.00
14	95.81
15	96.00
16	95.05
17	92.00

Embora a busca por uma leitura perfeita seja um objetivo contínuo, a taxa de 96% de acurácia com tolerância a dois caracteres demonstra que, para o contexto do trabalho, a solução proposta é plenamente funcional. Este desempenho agregado, especialmente quando contrastado com os trabalhos relacionados que frequentemente dependem de ambientes controlados, valida a abordagem como uma solução eficaz para a aplicação em campo.

5. Considerações Finais

Este trabalho apresentou o desenvolvimento e a avaliação de uma *pipeline* para a leitura automática de VINs gravados em superfícies metálicas, um desafio relevante para a automação industrial e o combate a fraudes. A solução foi estruturada em múltiplos estágios, compreendendo a detecção da região do chassi com OBBs, correção de orientação textual, detecção e classificação de caracteres individuais, e um refinamento final via pós-processamento baseado em regras de negócios.

Os resultados experimentais validaram a eficácia de cada componente da *pipeline*. A abordagem demonstrou alta robustez para aplicações práticas, alcançando 96% de acurácia na leitura completa do chassi ao se tolerar até dois caracteres incorretos. É importante ressaltar que, no contexto desta aplicação, a leitura não necessita ser infalível para agregar valor, podendo ser utilizada como um importante indicador no sistema de alerta para possíveis fraudes, que considera múltiplas variáveis para a tomada de decisão. Uma das principais contribuições do trabalho foi a avaliação do enriquecimento de dados de treinamento com imagens sintéticas. Esta estratégia foi importante para melhorar o desempenho do modelo de detecção de caracteres, que atingiu 99,72% de acurácia em um

	*	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	1	659	3	0	0	0	1	0	1	0	3	1	0	4	0	0	0	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	2	2	679	0	2	0	3	0	3	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	1	2	2	484	0	1	1	1	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	4	1	1	4	513	1	1	0	1	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	2	5	0	2	528	0	5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	1	1	3	2	0	1	565	2	1	0	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	0	4	1	0	1	1	1	392	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	1	16	2	1	0	4	1	332	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	
8	0	1	1	0	4	0	0	0	0	468	4	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	1	2	8	0	6	0	1	0	1	0	798	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
A	0	1	2	0	0	0	0	0	0	0	0	225	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B	0	1	0	0	0	0	2	1	0	1	1	0	608	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
C	0	0	0	0	0	0	0	0	0	0	0	0	0	127	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
D	0	1	0	0	0	0	0	0	0	0	0	0	0	0	184	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
E	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	87	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
F	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	170	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
G	0	3	1	0	0	0	0	4	0	1	0	0	0	0	0	0	0	267	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
H	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	219	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I	0	0	2	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	114	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
J	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	147	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
K	0	0	2	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	107	2	0	0	0	0	0	0	0	0	0	0	0	0	0	
L	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
M	0	0	0	0	0	0	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
N	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
T	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
U	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Z	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figura 5. Matriz de confusão para a classificação dos caracteres do chassi.

conjunto de testes real. Adicionalmente, a investigação sobre o pré-processamento dos recortes de caracteres revelou que a aplicação de um *padding* de 10% na caixa delimitadora antes da classificação foi crucial, elevando a precisão do classificador para 98%.

Apesar dos resultados promissores, a acurácia para a leitura perfeita (79,05%) ainda revela uma limitação e margem para aprimoramento. Essa lacuna, parcialmente atribuída à complexidade da *pipeline* sequencial, direciona trabalhos futuros para a exploração de estratégias de OCR mais holísticas, capazes de simplificar o processo e melhorar a compreensão contextual da ROI, visando aumentar a generalização do sistema. Outra limitação central do presente estudo é a ausência de uma análise de desempenho computacional. Dessa forma, uma continuação essencial será a avaliação rigorosa de métricas como latência de inferência e consumo de recursos (CPU/GPU), a fim de validar a viabilidade prática da solução para implementação em ambientes industriais com restrições de tempo real, um ponto não explorado no escopo atual.

Agradecimentos

Ao Laboratório de Aprendizado de Máquina Aplicado à Indústria - LAMIA*, pelo fomento à pesquisa e suporte. A Universidade Tecnológica Federal do Paraná, pelo apoio

*<https://www.lamia-edu.com>

fundamental e recursos disponibilizados para a realização deste trabalho de pesquisa.

Referências

- American Journal of Transportation (2024). Compliance and regulatory benefits of using vin decoders in international vehicle shipping. <https://www.ajot.com/news/compliance-and-regulatory-benefits-of-using-vin-decoders-in-international-vehicle-shipping>. Acessado em 2025-06-19.
- Associação Brasileira de Normas Técnicas (2001). NBR 6066:2001 - road vehicles - vehicle identification number (vin).
- Confederação Nacional das Seguradoras (2024). Quantificação da fraude no mercado de seguros brasileiro – relatório parcial 2024. <https://www.cnseg.org.br/publicacoes/22-ciclo-do-sqf-1-semester-de-2024>.
- Lubiato, K. (2015). Como as seguradoras combatem as fraudes em automóveis. <https://revistaapolice.com.br/2015/12/como-as-seguradoras-combatem-as-fraudes-em-automoveis/>. Acessado em 2025-08-03.
- Patil, A. V. and Dhanvijay, M. M. (2015). Engraved character recognition using computer vision to recognize engine and chassis numbers: Computer vision technique to identify engraved numbers. In *2015 International Conference on Information Processing (ICIP)*, pages 151–154. IEEE.
- Prakash, G., K, A., Maroof, M. M., Maria Fernandes, S., Roopashree, Bekal, A., and Shetty, R. G. (2023). Engraved character recognition using computer vision to recognize engine number with environmental condition setup. In *2023 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, pages 1–7.
- Ramshankar, Y. and Deivanathan, R. (2018). Development of machine vision system for automatic inspection of vehicle identification number. *International Journal of Engineering and Manufacturing (IJEM)*, 8(2):21–32.
- Shah, P., Karamchandani, S., Nadkar, T., Gulechha, N., Koli, K., and Lad, K. (2009). Ocr-based chassis-number recognition using artificial neural networks. In *2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, pages 31–34.
- Souza, L. R. d. S., Oliveira, R. M. M., Stoppa, M. H., and Caldas, J. O. (2014). Desenvolvimento de processo e dispositivo para inspeção da gravação de chassi utilizando visão de máquina. In *ABCM Symposium Series in Mechatronics - Vol. 6*, pages 1326–1333. ABCM.
- Terven, J., Cordova-Esparza, D.-M., Romero-González, J.-A., Ramírez-Pedraza, A., and Chávez-Urbiola, E. A. (2025). A comprehensive survey of loss functions and metrics in deep learning. *Artificial Intelligence Review*, 58(7).
- Yang, C.-H. and Feng, H.-S. (2020). One-stage vehicle engine number recognition system. In *2020 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*, pages 1–4. IEEE.