

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

ALISSON STOPASSOLE

**RECONHECIMENTO AUTOMÁTICO DE PLACAS
VEICULARES PARA O CONTROLE DE ESTACIONAMENTOS
BASEADO EM APRENDIZADO DE MÁQUINA E VISÃO
COMPUTACIONAL**

DOIS VIZINHOS

2020

ALISSON STOPASSOLE

**RECONHECIMENTO AUTOMÁTICO DE PLACAS
VEICULARES PARA O CONTROLE DE ESTACIONAMENTOS
BASEADO EM APRENDIZADO DE MÁQUINA E VISÃO
COMPUTACIONAL**

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do título de Bacharel em Engenharia de Software da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador: Prof. Dr. Yuri Kaszubowski Lopes

Coorientador: Prof. Dr. Gustavo Jansen de Souza Santos

DOIS VIZINHOS

2020



Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

TERMO DE APROVAÇÃO

TRABALHO DE CONCLUSÃO DE CURSO - TCC

RECONHECIMENTO AUTOMÁTICO DE PLACAS VEICULARES PARA O CONTROLE DE ESTACIONAMENTOS BASEADO EM APRENDIZADO DE MÁQUINA E VISÃO COMPUTACIONAL

Por
Alisson Stopassole

Monografia apresentada às 18 horas 30 min. do dia 23 de novembro de 2020 como requisito parcial, para conclusão do Curso de Bacharelado em Engenharia de Software da Universidade Tecnológica Federal do Paraná, Câmpus Dois Vizinhos. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação e conferidas, bem como achadas conforme, as alterações indicadas pela Banca Examinadora, o trabalho de conclusão de curso foi considerado APROVADO.

Banca examinadora:

Prof. Dr. Anderson Chaves Carniel	Membro
Prof. Me. Renê Pomilio de Oliveira	Membro
Prof. Dr. Yuri Kaszubowski Lopes	Orientador
Prof. Dr. Francisco Carlos Monteiro Souza	Professor(a) responsável TCCII



Documento assinado eletronicamente por (Document electronically signed by) **YURI KASZUBOWSKI LOPES, PROFESSOR DO MAGISTERIO SUPERIOR**, em (at) 04/12/2020, às 16:37, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por (Document electronically signed by) **ANDERSON CHAVES CARNIEL, PROFESSOR DO MAGISTERIO SUPERIOR**, em (at) 04/12/2020, às 20:27, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por (Document electronically signed by) **RENE POMILIO DE OLIVEIRA, PROFESSOR MAGISTERIO SUPERIOR-SUBSTITUTO**, em (at) 04/12/2020, às 20:55, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por (Document electronically signed by) **Alisson Anselmo Stopassole, Usuário Externo**, em (at) 04/12/2020, às 21:10, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site (The authenticity of this document can be checked on the website) https://sei.utfpr.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador (informing the verification code) **1792276** e o código CRC (and the CRC code) **559969A4**.

Dedico este trabalho a todas as pessoas que me auxiliaram de alguma forma na minha formação, de forma especial a todos meus professores pelo incentivo e conhecimento proporcionado e aos meus pais que mesmo com todas as dificuldades me deram suporte para continuar, e por último, mas não menos importante a minha avó, a qual convivi durante os anos acadêmicos e me proporcionou imenso conhecimento de vida.

AGRADECIMENTOS

Agradeço ao meu querido orientador e coorientador pelo aconselhamento para a construção deste trabalho. Estendo os agradecimentos a todos os professores e pessoas que de alguma forma contribuíram direta ou indiretamente para meu crescimento intelectual.

Agradeço a meus amigos, conquistados durante a jornada acadêmica, os quais por muitas vezes necessitei de ajuda e que também ajudei e foram essenciais para a conclusão das etapas da faculdade.

De forma especial agradeço e parablenizo a Universidade Tecnológica Federal do Paraná, pelo excelente conhecimento proporcionado a seus alunos, e pela dedicação na formação de profissionais qualificados.

O que as suas mãos tiverem que fazer, que o façam com toda a sua força, pois na sepultura, para onde você vai, não há atividade nem planejamento, não há conhecimento nem sabedoria.

Bíblia (Eclesiastes, 9:10)

RESUMO

STOPASSOLE, Alisson. RECONHECIMENTO AUTOMÁTICO DE PLACAS VEICULARES PARA O CONTROLE DE ESTACIONAMENTOS BASEADO EM APRENDIZADO DE MÁQUINA E VISÃO COMPUTACIONAL. 52 f. Trabalho de Conclusão de Curso – Coordenadoria do Curso de Engenharia de Software, Universidade Tecnológica Federal do Paraná. Dois Vizinhos, 2020.

Recentemente o Mercosul (Mercado Comum do Sul) adotou um padrão comum de placas veiculares que deverá substituir gradativamente as placas de cada um de seus estados membros. O presente trabalho destinou-se a implementar um algoritmo baseado em Inteligência Artificial e Visão Computacional capaz de identificar e reconhecer ambos, o novo padrão adotado de placas veiculares no formato Mercosul, bem como o formato brasileiro antigo. O objetivo deste trabalho foi realizar estudos com foco nas mudanças de placa e desenvolver um *framework* para a automatização do reconhecimento destas placas através de imagens e vídeos bem como demonstrar este *framework* em um protótipo de um sistema Web para controle de estacionamento. Para a validação, uma série de experimentos utilizando tanto placas antigas quanto novas foram conduzidos e resultados reportados.

Palavras-chave: Inteligência Artificial, Visão Computacional, Reconhecimento Automático de Placa, Placa Mercosul.

ABSTRACT

STOPASSOLE, Alisson. AUTOMATIC RECOGNITION OF PARKING CONTROL PLATES BASED ON MACHINE LEARNING AND COMPUTER VISION. 52 f. Trabalho de Conclusão de Curso – Coordenadoria do Curso de Engenharia de Software, Universidade Tecnológica Federal do Paraná. Dois Vizinhos, 2020.

Recently, Mercosul (Southern Common Market) adopted a common plate standard that should gradually replace the plates of each of its member states. This project aimed to implement an algorithm based on Artificial Intelligence and Computer Vision capable of identifying and recognizing both the new Mercosul plate standard adopted and the old Brazilian format. The objective of this work was to carry out studies focusing on the exchange of signs and to develop a web application that, together with the algorithm, can automate the parking control. For validation, a series of experiments using new and old plates were performed and the results reported.

Keywords: Artificial Intelligence, Computer Vision, Automatic Plate Recognition, Plate Mercosul.

LISTA DE FIGURAS

FIGURA 1	–	Formato placa Brasileira e placa Mercosul.	14
FIGURA 2	–	Etapas da fase de reconhecimento.	15
FIGURA 3	–	Mapa mental do documento.	16
FIGURA 4	–	Diagrama de UML da solução desenvolvida	18
FIGURA 5	–	Atividade realizadas durante este trabalho	26
FIGURA 6	–	Ilustração de Equalização do Histograma.	27
FIGURA 7	–	Imagem Original e Imagem Equalizada.	27
FIGURA 8	–	Distribuição dos Métodos de Obtenção de Imagens	30
FIGURA 9	–	Distância das Fotos	31
FIGURA 10	–	Ângulos das Fotos	31
FIGURA 11	–	Demonstração Placa Antiga	32
FIGURA 12	–	Demonstração Placa Mercosul	33
FIGURA 13	–	Demonstração da Detecção do Veículo	33
FIGURA 14	–	Demonstração das etapas de processamento da imagem da placa	34
FIGURA 15	–	Comparação de tempo Placa Antiga e Placa Nova	36
FIGURA 16	–	Matrizes	40
FIGURA 17	–	Treinamento 1	40
FIGURA 18	–	Resultados do Treinamento 1	41
FIGURA 19	–	Matrizes Treinamento 2	42
FIGURA 20	–	Treinamento 2	42
FIGURA 21	–	Resultados do Treinamento 2	43
FIGURA 22	–	Tela de cadastro dos veículos	43
FIGURA 23	–	Tela de listagem dos veículos	44
FIGURA 24	–	Tela de estacionamento	44
FIGURA 25	–	Tela de veículos no estacionamento	45

LISTA DE TABELAS

TABELA 1	– Fator de estudo:	36
TABELA 2	– Fator de estudo: Distância e resolução da imagem. Fotos tiradas durante o dia com câmera 2.	37
TABELA 3	– Fator de estudo: Distância e resolução da imagem. Fotos tiradas durante a noite com câmera 1.	37
TABELA 4	– Fator de estudo: Distância e resolução da imagem. Fotos tiradas durante a noite com câmera 2.	37
TABELA 5	– Fator de estudo: Ângulo e resolução da imagem. Fotos tiradas durante o dia com câmera 1.	38
TABELA 6	– Fator de estudo: Ângulo e resolução da imagem. Fotos tiradas durante o dia com câmera 2.	38
TABELA 7	– Fator de estudo: Ângulo e resolução da imagem. Fotos tiradas durante a noite com câmera 1.	38
TABELA 8	– Fator de estudo: Ângulo e resolução da imagem. Fotos tiradas durante a noite com câmera 2.	39

LISTA DE SIGLAS

UTFPR	Universidade Tecnológica Federal do Paraná
IA	Inteligência Artificial
DV	Dois Vizinhos
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
RNA	Rede Neural Artificial
MLP	Multilayer Perceptron
UML	Unified Modeling Language

SUMÁRIO

1	INTRODUÇÃO	12
1.1	MOTIVAÇÃO	13
1.2	OBJETIVO GERAL	13
1.3	OBJETIVOS ESPECÍFICOS	14
1.4	ESTRUTURA DA MONOGRAFIA	15
2	REVISÃO BIBLIOGRÁFICA	17
2.1	ENGENHARIA DE SOFTWARE	17
2.2	VISÃO COMPUTACIONAL	17
2.3	REDES NEURAIS	19
2.4	MOMENTOS INVARIANTES	19
2.5	OPERAÇÕES MORFOLÓGICAS	20
3	TRABALHOS RELACIONADOS	21
4	METODOLOGIA	25
4.1	FUNDAMENTOS	25
4.2	TECNOLOGIAS	27
4.3	DESCRIÇÃO DOS TESTES	28
4.4	RECURSOS	28
5	RESULTADOS	30
5.1	AQUISIÇÃO DE DADOS PARA O EXPERIMENTO	30
5.2	PROCESSAMENTO DO ALGORITMO DE RECONHECIMENTO	32
5.3	RESULTADOS FINAIS	34
6	CONSIDERAÇÕES FINAIS	46
	REFERÊNCIAS	48
	Anexo A – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO	51

1 INTRODUÇÃO

Considerando o aumento da frota veicular no Brasil e no mundo, o reconhecimento automático de placas veiculares tornou-se essencial para o controle e segurança do trânsito e de seus proprietários (GUINGO; RODRIGUES; THOMÉ, 2002). O reconhecimento inteligente de caracteres alfanuméricos de imagens digitalizadas de veículos tem diversas aplicações, como por exemplo no setor comercial: (i) o controle de portões ou cancela de garagens e estacionamentos, (ii) o controle de acesso, do tempo de uso e cobrança de vagas, e (iii) a administração da quantidade de vagas disponíveis, sem a necessidade de um trabalhador para realizar este serviço, eliminando também falhas humanas (FERREIRA; SOARES, 2012). Além das aplicações de cunho comercial, tem-se também uso para aplicação da Lei e de cumprimento da ordem de trânsito, através do (iv) monitoramento de veículos furtados ou irregulares transitando em via pública, (v) a identificação de infrações de trânsito, e a (vi) identificação de veículos utilizados em crimes (LAXMI; ROHIL, 2014).

Um exemplo de reconhecimento de placas é o sistema de radares do Departamento de Trânsito (DETRAN), popularmente conhecido como “pardais”, responsável por monitorar a velocidade, e que realiza a aplicação de multas a veículos com velocidade maior que a permitida na via, através do reconhecimento dos caracteres presentes nas placas. A automatização do processo de reconhecimento pode ser obtida com Inteligência Artificial (IA), que é um ramo de estudo da Ciência da Computação que visa desenvolver aplicações e/ou dispositivos que possam simular a forma exclusiva de pensar e agir com racionalidade e inteligência do ser humano (TEIXEIRA, 2019).

Apesar da existência de técnicas automáticas de reconhecimento de placas, recentemente um novo padrão de placas foi adotado pelo Brasil, o padrão Mercosul (Mercado Comum do Sul), que consiste na padronização de placas para facilitar a identificação de veículos nos países-membros deste acordo, formado atualmente pelo Brasil, Argentina, Paraguai e Uruguai (MERCOSUL, 2019).

Neste trabalho são apresentados alguns impactos encontrados na diferenciação

entre as duas placas, sendo um deles o fato de a placa Mercosul ser menos refletiva que a placa antiga. Assim como foram elaborados os processos de identificação da placa, recorte e reconhecimento dos caracteres, além de estudos e resultados de impactos no reconhecimento de placas veiculares expostos a fatores tais como luz, distância e ângulos de captura.

1.1 MOTIVAÇÃO

Tradicionalmente as placas veiculares são identificadas através de equipamentos distribuídos em vias ou locais privados, os quais realizam a captura das imagens dos veículos que trafegam; estas imagens são enviadas para um setor responsável pelo monitoramento do local, que então faz a identificação das placas. Entretanto, o maior problema desta abordagem é que ela é custosa, manual, tediosa e arcaica.

Como mostra a Figura 1, o novo padrão adotado diverge do antigo, não sendo mais formado por três letras seguidas por quatro algarismos (*e.g.*, AAA-XXXX), mas por uma sequência aleatória de sete caracteres alfanuméricos (*e.g.*, AAAXAXX). Atualmente a placa é refletiva, com fundo branco, com o nome da cidade e a identificação do estado. No entanto, a nova placa padrão Mercosul altera alguns elementos, tais como o topo da placa que recebe uma faixa azul de lado a lado e o nome do país de registro, substituindo o nome da cidade e estado. Também foi adicionado o desenho da bandeira do país de registro à direita, a bandeira do Mercosul à esquerda e brasão do município com nome por extenso (FORMATO, 2019) juntamente com um *QR code* de identificação. Na Figura 1, pode-se observar as mudanças nas placas tanto de carro como de motocicleta no formato antigo seguidos respectivamente pelo padrão Mercosul (PLACA, 2019).

Assim, a motivação foi identificar e estudar os impactos deste novo padrão adotado no formato de placas, pois tais mudanças precisam ser acompanhadas para propor uma solução capaz de automatizar, assim como no formato antigo, o processo de reconhecimento.

1.2 OBJETIVO GERAL

O trabalho destinou-se a implementar um algoritmo capaz de identificar e reconhecer o novo padrão adotado de placas veiculares no formato Mercosul, bem como o formato brasileiro antigo. Um estudo de caso foi desenvolvido utilizando-se de tecnologias de desenvolvimento atuais com grande potencial emergente, a fim de desenvolver uma aplicação Web que consiga automatizar o controle de estacionamentos utilizando nesta os princípios



Figura 1: Formato placa Brasileira e placa Mercosul.

Fonte: Pinterest (Adaptado)

da Engenharia de Software.

A fase de reconhecimento foi dividida em três etapas, sequencialmente ilustradas na Figura 2. A etapa inicial consistiu no monitoramento do fluxo do trânsito identificando o veículo e a captura da imagem das placas do mesmo. Em seguida, a imagem capturada é catalogada entre nova ou antiga e processada por um algoritmo treinado com uma base de dados, a fim de reconhecimento de seus caracteres. Por fim o resultado obtido na saída do processamento é destinado para uso no software de controle de estacionamento, o qual tem aplicações como descritas na Introdução deste trabalho.

1.3 OBJETIVOS ESPECÍFICOS

Os seguintes pontos foram estipulados para que houvesse o sucesso do estudo realizado:

- Identificar o veículo em vídeo;
- Identificar a placa no *frame* contendo a imagem de carro;
- Identificar o padrão da placa diferenciando a mesma entre formato antigo e formato Mercosul;

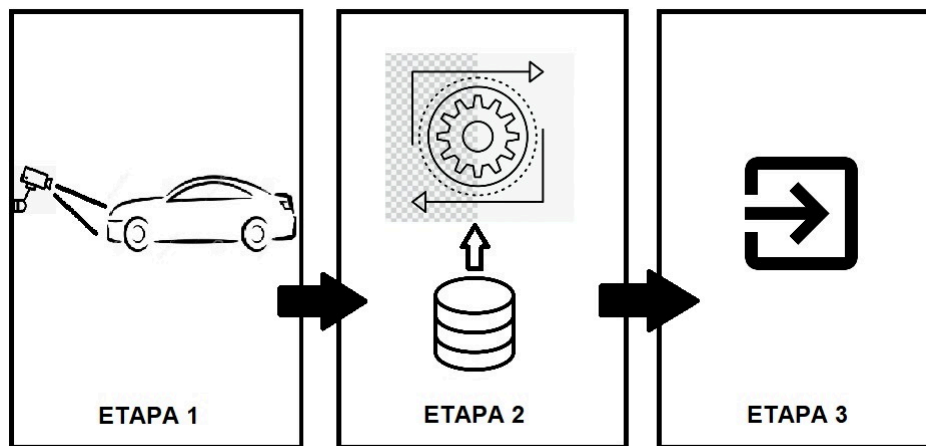


Figura 2: Etapas da fase de reconhecimento.

Fonte: Pinterest (Adaptado).

- Realizar experimentos comparando as diferenças entre as duas placas;
- Desenvolver o reconhecimento capaz de encontrar os caracteres;
- Desenvolver a aplicação Web utilizando a Engenharia de Software para interagir com o algoritmo de reconhecimento;
- Realizar experimentação;
- Compreender e interpretar os resultados obtidos para melhorá-los;

1.4 ESTRUTURA DA MONOGRAFIA

Este documento está dividido em 6 capítulos. Neste primeiro capítulo (Introdução) é apresentado as aplicações e soluções possíveis com o trabalho desenvolvido, assim como os objetivos e estrutura do mesmo. No Capítulo 2 é apresentada uma revisão bibliográfica de estudos e conhecimentos que foram utilizados no decorrer do trabalho, que assim como o Capítulo 3 serviram de base e inspiração por apresentarem conceitos e ideias que poderiam ser utilizadas e/ou reaproveitadas.

No Capítulo 4 são descritas quais ações foram tomadas e as possíveis soluções a serem adotadas, além de recursos e tecnologias estudados e utilizados para que no Capítulo 5 fossem apresentados os resultados de todo o trabalho, as soluções encontradas, sucessos e falhas obtidos, descrevendo o passo a passo do algoritmo e de como foi conduzido a aquisição de dados da experimentação.

Por fim as considerações do trabalho foram elaboradas, deixando explícito possíveis melhorias e trabalhos futuros, e explicando porque algumas partes não foram executadas. Foram adicionadas todas as fontes e referências de estudos utilizadas, assim como os documentos assinados pelas pessoas que colaboraram na elaboração deste trabalho final. Todo o processo pode ser observado no mapa mental da Figura 3;

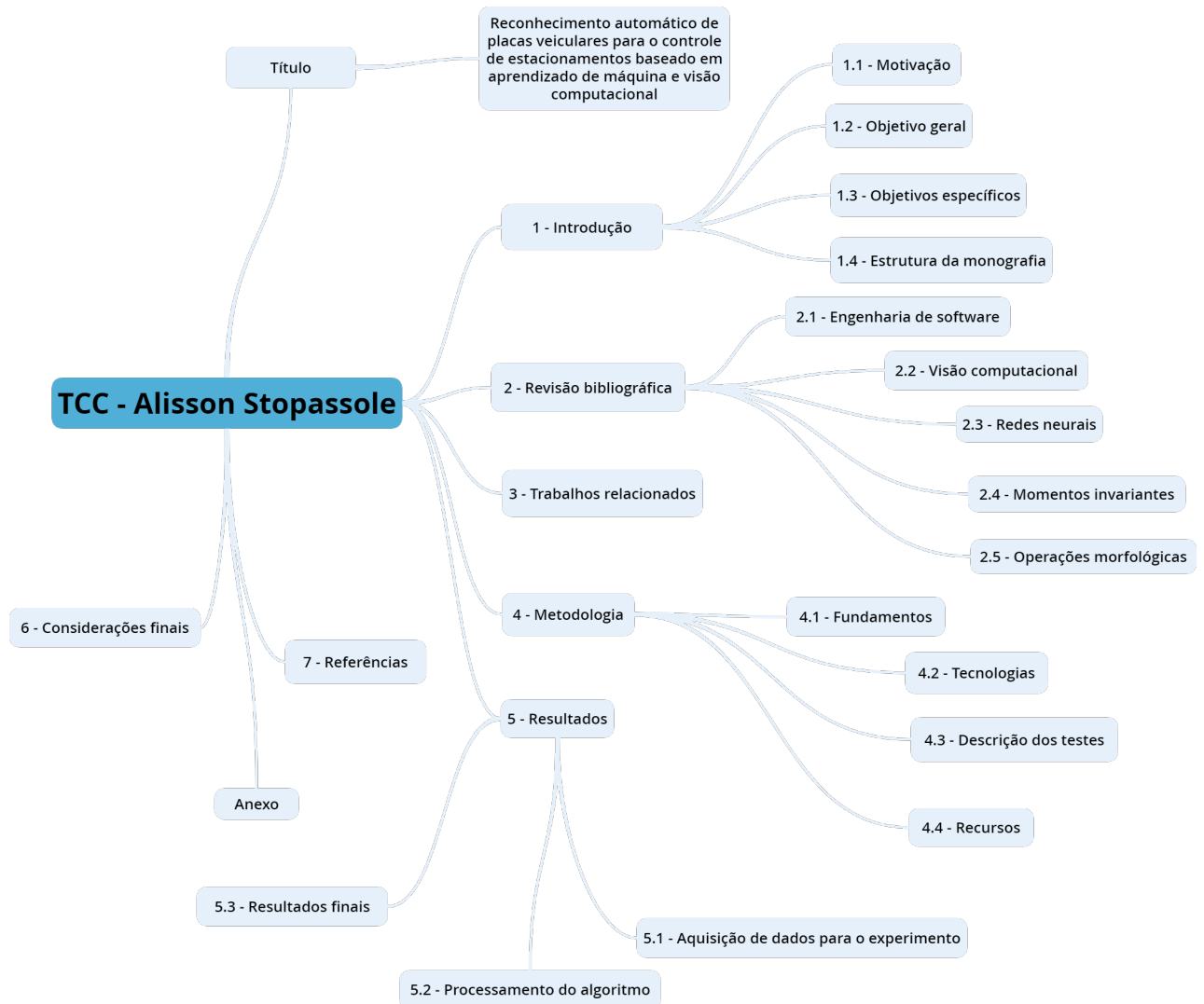


Figura 3: Mapa mental do documento.

Fonte: Autoria própria.

2 REVISÃO BIBLIOGRÁFICA

Nesta seção estão descritos os conceitos relacionados para a compreensão e o desenvolvimento deste trabalho. Todas as decisões tomadas para a execução deste trabalho foram norteadas por este levantamento bibliográfico, que contém o embasamento técnico-científico necessário para este estudo.

2.1 ENGENHARIA DE SOFTWARE

Segundo PRESSMAN (2011), a Engenharia de Software “é o estabelecimento e o emprego de sólidos princípios de engenharia de modo a obter software de maneira econômica, que seja confiável e funcione de forma eficiente em máquinas reais”. Ainda segundo PRESSMAN (2011), o Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) define como “a aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, na operação e na manutenção de software”.

Neste contexto, diversos métodos são aplicados ao processo de produção de um software para que se alcance um objetivo primordial, que é entregar dentro do prazo, com alta qualidade, o software operacional contendo funções e características que satisfaçam as necessidades de todos os envolvidos. As principais áreas relativas à Engenharia de Software de interesse deste trabalho são a Codificação (Programação) e a Modelagem de Sistemas, por meio de diagramas da *Unified Modeling Language* (UML), e esta foi empregada na diagramação da solução desenvolvida como pode-se ver na Figura 4.

2.2 VISÃO COMPUTACIONAL

Dentre várias subáreas de estudos da IA temos a Visão Computacional, que tem como objetivo replicar a capacidade da visão humana de extrair dados e ou informações de imagens ou quaisquer dados multidimensionais, através de padrões e características presentes nos mesmos (BACKES; JUNIOR, 2019). Mesmo que discretamente, aplicações

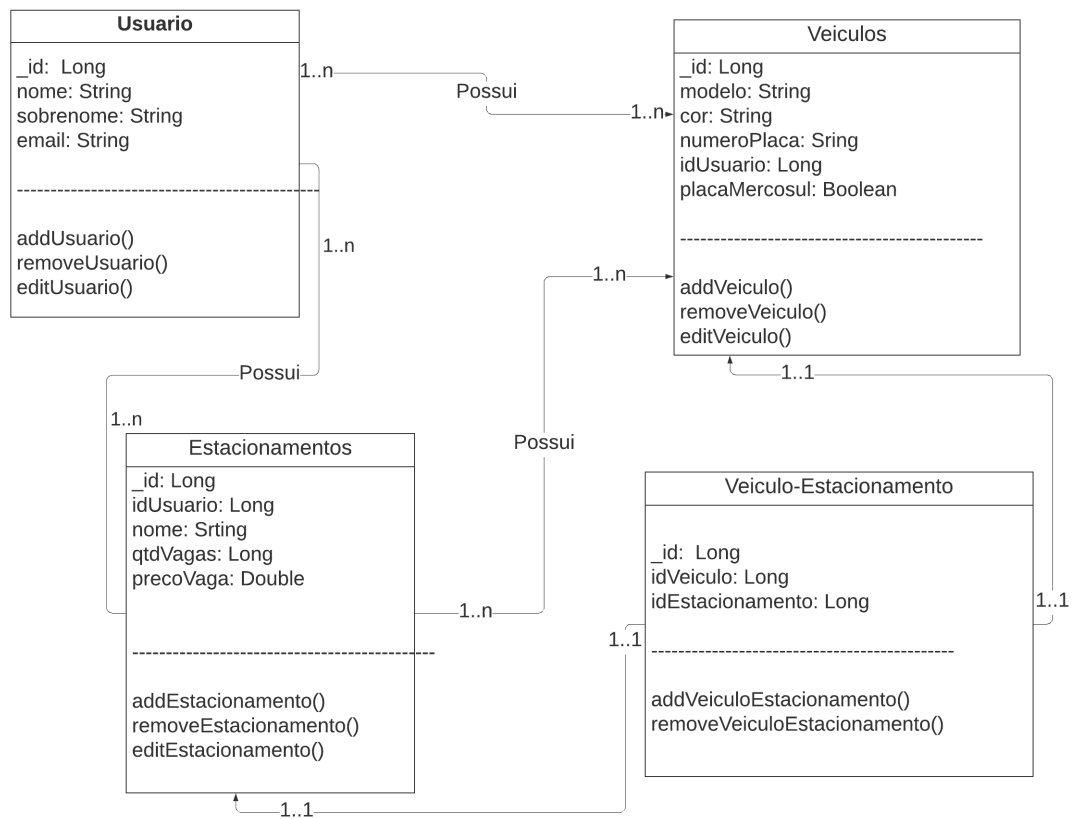


Figura 4: Diagrama de UML da solução desenvolvida

Fonte: Autoria Própria

que utilizam IA juntamente com Visão Computacional estão presentes no dia-a-dia, está em nossas câmeras, reconhecendo rostos e sorrisos. Está presente em carros autônomos, leitura de sinais de trânsito e reconhecendo pedestres, entre outras aplicações (ACADEMY, 2019).

Neste trabalho algumas técnicas foram aplicadas, a de redimensionamento da imagem foi uma desta, que consiste em aumentar ou diminuir o tamanho da imagem, este procedimento pode afetar a qualidade da imagem, para solucionar este impacto uma segunda técnica deve ser aplicada, sendo esta a de normalização da imagem que consiste em deixar a imagem com os *pixels* melhor distribuídos, melhorando a qualidade da imagem (ANTONELLO, 2014).

Uma terceira técnica utilizada foi o processo de binarização da imagem, que consiste em transformar a imagem para cores binárias, ou seja, preto e branco, ou em outras cores específicas. Com a finalidade de identificar o formato de uma placa veicular para poder reconhecer seus dados, e detectar tons de azul, este trabalho estudou a Visão

Computacional no reconhecimento de formas e cores em imagens.

2.3 REDES NEURAIAS

A pesquisa de Redes Neurais Artificiais (RNA) tenta simular o funcionamento da inteligência residente nos neurônios humanos e mapeá-la para uma estrutura artificial, por exemplo uma combinação de hardware e software, assim simulando as redes neurais biológicas em redes neurais artificiais (RAUBER, 2005).

A RNA a ser desenvolvida para este trabalho trata-se de uma *Multilayer Perceptrons* “MLP“. Uma MLP é uma rede neural artificial composta por mais de um *Perceptron*. O *Perceptron*, pode aprender apenas padrões linearmente separáveis, já o MLP pode aprender padrões não linearmente separáveis, é um algoritmo simples destinado a realizar a classificação binária classificando se a entrada pertence ou não a determinado grupo, validando e comparando até encontrar um resultado. O MLP é composto por uma camada de entrada para receber o sinal, uma camada de saída que toma uma decisão ou previsão sobre a entrada (BOOK, 2019).

A RNA foi desenvolvida para o reconhecimento dos caracteres, a aprendizagem da mesma se deu por meio supervisionado onde cada exemplo de treino foi acompanhado por um valor que é o valor desejado, quando o resultado esperado não foi adequado uma adaptação foi aplicada e novamente realizado o treinamento.

2.4 MOMENTOS INVARIANTES

A técnica de momentos invariantes (ou apenas momentos), apesar de pouco aplicada, é uma alternativa para o reconhecimento, também chamada de momentos estatísticos, é um dos vários métodos utilizados para extração de características de uma imagem. Estes momentos e as funções derivadas deles se caracterizam por valores numéricos calculados a partir da imagem previamente segmentada e que descrevem a distribuição espacial dos pontos contidos na imagem ou em uma região (CHACON et al., 2011).

Momentos invariantes ou Momentos de HU, em homenagem a seu criador Ming-Kuei Hu, são uma combinação representadas por sete equações de momentos que são invariantes à escala, rotação e translação, responsável pela formação do vetor de características dos objetos.

2.5 OPERAÇÕES MORFOLÓGICAS

O princípio da morfologia matemática consiste em extrair informações referentes à geometria e à topologia de uma imagem digital. Uma operação morfológica é determinada a partir de uma vizinhança ao redor do ponto central de um *pixel*. A grande potencialidade da morfologia matemática reside nos elementos estruturantes, que são conjuntos geométricos definidos e conhecidos em forma e tamanho. Os elementos estruturantes são comparados, a partir de uma transformação, ao conjunto desconhecido da imagem. O resultado dessa transformação permite avaliar o conjunto desconhecido. O formato e o tamanho do elemento estruturante possibilitam testar e quantificar de que maneira, o elemento estruturante está ou não está contido na imagem (BUSSADOR, 2009).

As operações morfológicas são transformações que resultam na alteração da forma da imagem, através da manipulação de seus *pixels*. As operações morfológicas aplicam um operador sobre a imagem original, o resultado é uma imagem com o mesmo tamanho da original, mas com seus *pixels* melhor distribuídos por toda a imagem. Estas operações, por si só, causam distorções nas áreas dos objetos. A combinação dos operadores morfológicos gera resultados interessantes.

3 TRABALHOS RELACIONADOS

Nesta revisão narrativa, dentre todos os estudos realizados, não foi encontrado nenhum estudo referente ao reconhecimento do padrão Mercosul, sendo este um dos objetivos do trabalho, e o principal diferencial, além de que, o presente trabalho aplicou uma gama de testes na experimentação, caso que não é observado nos trabalhos relacionados.

Os seguintes trabalhos foram obtidos através de uma Revisão Narrativa, com o intuito de entender, como desenvolver e quais suas utilidades em âmbitos gerais de uma aplicação de reconhecimento automático de placas. Para a seleção destes foi analisado o conteúdo do mesmo, as técnicas aplicadas e por fim o procedimento da experimentação.

Apesar de alguns estudos obterem uma porcentagem de acerto de 100% ou próximo no reconhecimento, vários estudos não levaram em consideração variáveis de impacto que poderiam ser testadas, tais como iluminação noturna, assim como distâncias maiores, ângulos distintos, que poderiam ser aplicadas para melhor validação da capacidade das aplicações desenvolvidas.

Guingo, Rodrigues e Thomé (2002) descrevem uma estratégia adotada para a concepção e desenvolvimento de um sistema para reconhecimento automático de placas de veículos automotores utilizando duas redes neurais do tipo *Feedforward Multi-Layer Perceptron*, treinadas com o algoritmo *backpropagation*. Seu trabalho foi modularizado em 6 etapas, sendo elas: Obtenção da fotografia do veículo; Localização da região da placa; Seleção e extração da região da placa; Segmentação dos caracteres da placa; Extração das características dos caracteres segmentados e Reconhecimento dos caracteres. Os autores concluíram que os resultados estão abaixo do esperado devido à baixa qualidade das imagens obtidas em situação real de vias públicas, tendo uma estimativa de 76.61% de acerto no módulo de segmentação utilizando uma base de 5.233 fotos. Os autores ressaltaram a dificuldade no reconhecimento das letras “B”, “D” e “O” e os algarismos “0” e “8”.

Ferreira e Soares (2012) desenvolveram um algoritmo que percorre todos os *pixels* da imagem original analisando se a tonalidade tem a possibilidade de ser uma placa de cor cinza. Caso esta não seja encontrada o processo é repetido para vermelho, branco, preto e azul. Quando a imagem é encontrada, os caracteres são segmentados e inseridos em uma rede neural para processamento. O sistema foi testado com placas de veículos em imagens adquiridas de 3 a 7 metros de distância, providas de iluminação natural. A câmera foi posicionada entre 1,5 e 2,5 metros de altura em relação ao piso do veículo. Os veículos estavam em rotação de até 15 graus, resultando em um percentual de 100% de acerto nos testes e condições aplicados.

Conci e Monteiro (2004) também desenvolveram um sistema composto de etapas. Na primeira etapa a imagem é adquirida. Na segunda, a imagem é pré-processada, visando identificar a região de possível placa nas imagens. Na terceira, só a região da placa é processada visando obter características de cada caractere que, uma vez extraídas, possibilitem sua análise. Finalmente os caracteres são classificados para posterior identificação da placa.

Os autores destacaram a dificuldade no reconhecimento dos números 7 e 3, assim como nas letras, os caracteres “H” e “N” bem como “D” e “O”. Mas sempre, apesar de alguns momentos invariantes apresentarem problemas no reconhecimento de alguns caracteres, os outros momentos foram capazes de reconhecer com precisão o mesmo. Esse fato assegurou um reconhecimento de 100% dos caracteres.

Foram processados 217 caracteres de 31 placas de automóveis, e reconhecidos 112 caracteres sem a necessidade de utilização dos momentos invariantes. Sendo assim 105 caracteres foram reconhecidos utilizando a técnica dos momentos invariantes. Foram realizados 505 cálculos de momentos onde apenas 49 apresentaram erro de reconhecimento. Esses erros não prejudicaram o reconhecimento como um todo, pois 100% das placas processadas foram reconhecidas com sucesso.

Fontana e Marim (2010) implementaram uma rede neural artificial do tipo *feedforward* devido ao algoritmo de *backpropagation*. Essa rede foi desenvolvida somente para reconhecer os números pois não mostrou eficiência para o reconhecimento das letras. A rede neural foi treinada com um total de duzentas imagens.

Pelo fato de serem sete termos que adquirimos da placa, para cada caractere, o algoritmo faz a varredura da imagem com um passo que determina a possível variação da posição do mesmo em relação à placa. Isso se mostrou como uma solução para imperfeições na imagem, ângulo de inclinação da placa e também os efeitos da luminosidade. Assim,

imagens em que a foto não conseguiu focar a placa do carro ou apresentou um tamanho diferenciado devido ao ângulo de inclinação, podem ser reconhecidas e ter os caracteres extraídos.

Após desenvolver o algoritmo capaz de identificar as placas nas imagens obtidas, a taxa de reconhecimento das placas foi de 99%, sendo que dentre as placas não reconhecidas havia imagens de automóveis cujas placas estavam deterioradas, a ponto de não serem reconhecidas nem pelo olho humano.

Fonseca-Galindo, Castro e Braga (2016) propuseram um sistema contendo três módulos principais: o primeiro módulo é responsável pela localização da placa a partir da aplicação de um conjunto de operadores morfológicos. O segundo módulo faz a segmentação dos caracteres da placa com base na sua projeção horizontal e vertical. Finalmente, o terceiro módulo realiza o reconhecimento de caracteres com classificadores *Multilayer Perceptron*.

O classificador conseguiu reconhecer 37 de 43 placas, alcançando um reconhecimento das placas de 76%. Os autores ressaltaram, no entanto, que o sistema logrou reconhecer 165 de 172 caracteres numéricos, um total de 96% de reconhecimento para números.

Neto, Filho e Norte (2013) utilizaram uma Rede Neural Artificial do tipo *Multilayer Perceptron* para o reconhecimento dos caracteres. Sua aplicação utilizou 700 vídeos e a partir destes foram extraídas 12000 imagens. Destas imagens, foram extraídos corretamente 47808 números de um total de 48000, obtendo assim uma taxa de 99,6% e 35895 letras de um total de 36000, obtendo assim uma taxa de 99,71% nas letras, resultando em uma taxa de extração de 97,02%.

Na etapa de reconhecimento, do total de 47808 números, 46751 foram reconhecidos, obtendo uma taxa de acerto de 97,7%. Quanto às letras, a etapa do reconhecimento teve como entrada um total de 35895, acertando um total de 34618 caracteres e atingindo uma taxa de acerto de 96,4%. Com isso, o sistema final possui uma taxa de acerto de aproximadamente 94%.

Bussador (2009) utilizou uma abordagem menos convencional. Na realização do trabalho foi necessária a criação de uma base de dados, contendo 180 imagens digitais de veículos estacionados e no desenvolvimento aplicou a utilização de ferramentas de morfologia matemática, tais como granulometria em níveis de cinza e *tophat*, ou seja, conversão da imagem para escalas de cinza, para o processamento das imagens no intuito de segmentar e localizar as placas dos veículos.

Em uma base de 60 imagens, 75% de acerto na localização das placas e 81% de acerto na segmentação dos caracteres corretamente localizados, podem ser considerados como satisfatórios levando em consideração as diferentes situações das imagens analisadas, levando-se em consideração a complexidade da imagem e a não utilização de algoritmos clássicos tais como Redes Neurais para a resolução de problemas que envolvem este nível de complexidade.

Dentre todos os trabalhos citados, podemos observar a predominância do uso de Redes Neurais, e ainda que não citado explicitamente, houve uma grande utilização de técnicas de pré-processamento de imagens, utilizadas para se obter maior eficiência no processo de identificação das placas, e reconhecimento dos caracteres das mesmas, sendo estas as duas principais motivações e concordâncias encontradas e que foram aplicadas na prática no trabalho.

É evidente também a diferença entre os resultados obtidos, tendo uma grande variância de percentual de acerto, ou diferença nos tipos de análise de resultados, além de diferença de ambiente de testes, sendo que muitos testes foram realizados em ambientes simulados e não se utilizando de testes reais, tais como um radar de trânsito ou um estacionamento.

4 METODOLOGIA

O trabalho foi dividido em duas fases, as quais são complementares uma a outra para que o objetivo final seja alcançado. A primeira fase consistiu em desenvolver um algoritmo de reconhecimento de placas para a placa cinza tradicional e placa no formato Mercosul, enquanto a segunda teve por finalidade desenvolver o software que utilize o algoritmo desenvolvido em uma aplicação Web para gerir estacionamentos.

A fase inicial é composta por: Uma imagem inicial contendo um veículo posicionado de frente ou de traseira é inserida no algoritmo. O algoritmo então faz o recorte da área com a placa. Após o recorte, a sub-imagem é direcionada ao processamento para ser classificada em placa nova ou placa antiga. Depois de classificada, a mesma imagem é redirecionada para os reconhecimentos dos caracteres.

A segunda fase consistiu em integrar todas as partes do algoritmo, responsável pelo monitoramento do veículo, obtenção da foto, recorte da placa, obtenção dos caracteres reconhecidos na etapa inicial para que o usuário possa interagir por meio da aplicação Web. Pode-se observar o processo no diagrama de atividades da Figura 5

O algoritmo responsável pelo recorte da área com a placa e diferenciação entre nova e antiga passou por uma experimentação cujos resultados serão reportados no Capítulo 5, pois trata-se de uma etapa crítica do reconhecimento. Da mesma forma, toda a aplicação passou por um segundo instante de experimentação, onde foram testadas todas as partes integradas e quais os possíveis impactos da integração, assim como os dados de porcentagem de acerto e de erro.

4.1 FUNDAMENTOS

O algoritmo utilizado e observado como mais adequado para a primeira fase do desenvolvimento foi o de Viola-Jhones assim nomeado em homenagem a seus criadores Paul Viola e Michael Jones, também conhecido pelo nome de *Haar-Cascade* (OpenCV, 2019a). Este algoritmo é uma abordagem baseada em aprendizado de máquina e muito utilizado

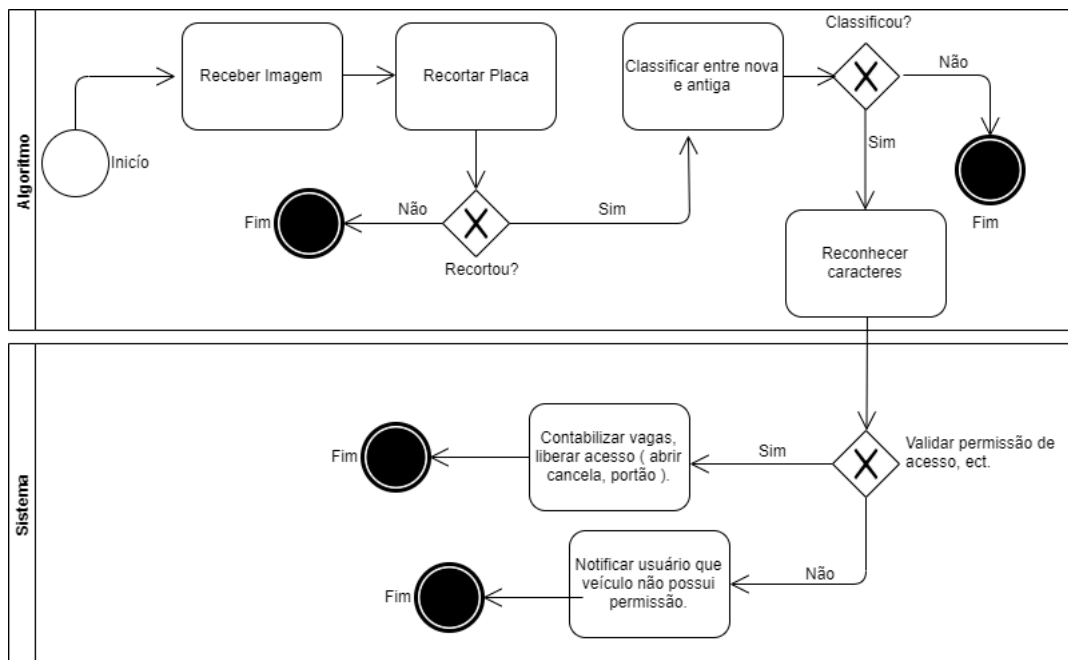


Figura 5: Atividade realizadas durante este trabalho

Fonte: Autoria Própria

para detecção de contornos de objetos. Ele basicamente necessita de um treinamento com um conjunto de imagens que possuem características que devem ser reconhecidas pelo algoritmo e um conjunto de imagens que não possuem características que devem ser reconhecidas.

Outro processo utilizado é o de Equalização do Histograma, que tem como finalidade melhorar o contraste em uma imagem, no algoritmo foi aplicado para imagens muitas pequenas para facilitar seu reconhecimento. Por exemplo, uma imagem mais clara terá todos os *pixels* agrupados porém, uma boa imagem terá *pixels* de todas as regiões da imagem. Portanto, é necessário expandir esse histograma para qualquer uma das extremidades, fazendo com que seus *pixels* sejam melhor dispersos por toda a área da imagem, como mostrado nas Figuras 6 e 7. Esse processo normalmente melhora o contraste da imagem (OpenCV, 2019c), pois aproveita toda a faixa de cores disponíveis.

Utilizou-se também a transformação de imagens com determinado espaço de cores para outra cor. Essas transformações são aplicadas pois as imagens originais precisam ser processadas em tons de cinza para otimização do processo (OpenCV, 2019b). Uma aplicação alternativa foi a transformação para tons de azul presentes na placa Mercosul, a fim de diferenciar a placa nova da placa antiga, onde as cores diferentes de azul são removidas e os tons de azul são ressaltados.

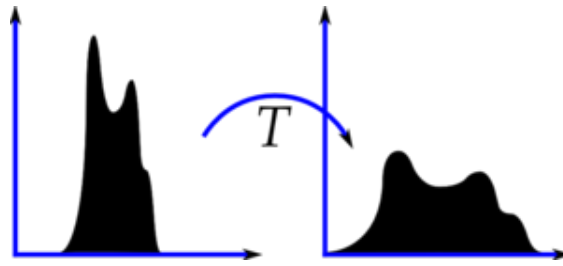


Figura 6: Ilustração de Equalização do Histograma.

Fonte: OpenCV

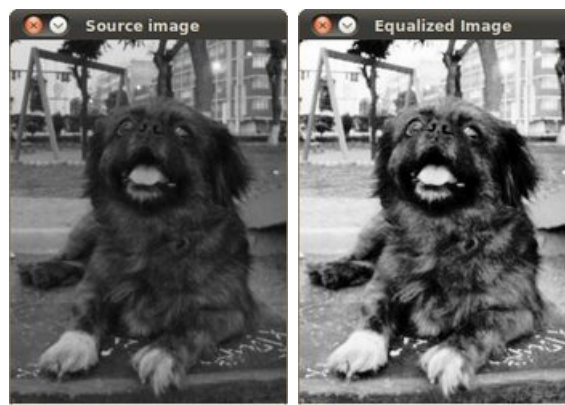


Figura 7: Imagem Original e Imagem Equalizada.

Fonte: OpenCV

Para o reconhecimento dos caracteres da placa foram utilizadas redes neurais convolucionais (CNN), que é inspirada no processo biológico de processamentos de dados visuais. De maneira semelhante aos processos tradicionais de visão computacional, uma CNN é capaz de aplicar filtros em dados visuais, mantendo a relação de vizinhança entre os *pixels* da imagem ao longo do processamento (VARGAS; PAES; VASCONCELOS, 2016). A rede neural tem como funcionalidade processar e inferir informações a partir de vários dados de entradas a fim de encontrar um resultado esperado.

4.2 TECNOLOGIAS

A tecnologia utilizada para desenvolvimento do algoritmo foi Python por conter uma rica coleção de bibliotecas para trabalhar com imagens e processamento de dados, mais especificamente com foco em bibliotecas que contemplem Visão Computacional, Linguagem de Máquina e processamento de imagens (MENEZES, 2010; MITCHELL, 1997; COIMBRA, 2004; VINHAES, 2013; PEDREGOSA et al., 2011).

Por outro lado, para a aplicação Web utilizou-se a linguagem de programação JavaScript, por ser uma linguagem em alta no mercado e pela opção de ser trabalhada tanto do lado cliente como do lado servidor na mesma aplicação. No *back-end*, aplicou-se o interpretador Node.js e no *front-end*, Angular juntamente com *Bootstrap*, *JQuery* e outros *frameworks* que podem vir a ser necessários. Na parte de banco de dados, utilizou-se NoSQL MongoDB, por ser altamente escalável e de fácil utilização, e por permitir trabalhos sem utilização de tipagem e *Structured Query Language* (SQL) (SILVA, 2010; TEIXEIRA, 2012; EDMONDS, 1996; CHODOROW, 2013).

4.3 DESCRIÇÃO DOS TESTES

Fase 1: Para a realização do experimento do algoritmo responsável pelo recorte da área com a placa, processamento da imagem e diferenciação entre nova e antiga, foram utilizados dois carros, um contendo placa antiga e o outro a placa nova. Para cada veículo foram tiradas uma série de fotos, todas no mesmo ambiente mas sujeitas a interferências externas, tais como: distância, ângulo, resolução e luminosidade. As imagens foram então submetidas ao processamento passando pelas etapas ilustradas no Capítulo 5, e seus resultados foram tabelados, para final comparação de resultados.

Fase 2: Uma segunda parte do algoritmo foi desenvolvida, esta parte responsável pela detecção do veículo em vídeo e pelo reconhecimento dos caracteres. Todo o código contendo detecção do veículo em vídeo, segmentação da região da placa, reconhecimento dos caracteres e plataforma WEB foi integrado e testado, visando calcular sua eficiência no reconhecimento e no gerenciamento do estacionamento.

Para a finalização do algoritmo foram necessárias as implementações de duas redes neurais uma para letras e outra para números, para a execução do treinamento das mesmas foram extraídas imagens internet e adicionadas a base de treino. A imagem precisa ser dimensionada a altura máxima de 128 *pixels*, e estar de acordo com as fontes utilizadas nas placas, sendo estas a fonte *Mandatory* da placa antiga e a fonte *FE Engschrift* da placa Mercosul.

4.4 RECURSOS

Todos os recursos utilizados são de propriedade própria e/ou emprestados com consentimento de colegas e professores e todas as tecnologias de desenvolvimento utilizadas são de código aberto. Os veículos utilizados nas fotografias das placas, são de origem

de terceiros. Ambos os proprietários, assinaram um termo de livre consentimento que consta no Anexo A. O algoritmo responsável pelo recorte das placas foi implementado, no entanto seu arquivo com o devido treinamento advém de um repositório de código aberto do GitHub, disponível em <https://github.com/openalpr/openalpr>.

As redes neurais convolucionais utilizadas, foram adaptadas de um código disponível na Internet, criada e disponibilizada pelo Professor André Gustavo Schaeffer da Universidade Federal da Fronteira Sul Câmpus Erechim, onde utiliza as mesmas em um contexto de reconhecimento de dígitos manuscritos. O material pode ser encontrado no endereço https://www.youtube.com/watch?v=uTeCEB3-_KE, juntamente com outros projetos.

5 RESULTADOS

5.1 AQUISIÇÃO DE DADOS PARA O EXPERIMENTO

Para a primeira fase da experimentação um *dataset* com um total de 192 imagens foi criado, fotografadas com duas câmeras de resoluções distintas da posição dianteira de dois veículos, um com placa Mercosul e outro com placa antiga. A câmera 1 foi a de um celular Samsung Galaxy S8 com 12 megapixels e uma resolução de 4290x2800 *pixels*. A câmera 2 foi uma câmera Sony com 14.2 megapixels e uma resolução 4592x3056. As mesmas foram posicionadas a aproximadamente 1,60 metros de altura em relação ao chão.

Cada câmera foi responsável pela captura de um total de 96 imagens, 50% das mesmas foram tiradas durante o dia e os outros 50% durante a noite com iluminação pública, no dia 23 de outubro de 2019, subdivididas entre placa nova e placa antiga, obedecendo aos parâmetros de teste adicionais de distância e ângulo, que resultaram em 15 e 9 imagens respectivamente. Podemos observar a disposição das imagens para a execução do experimento na Figura 8.

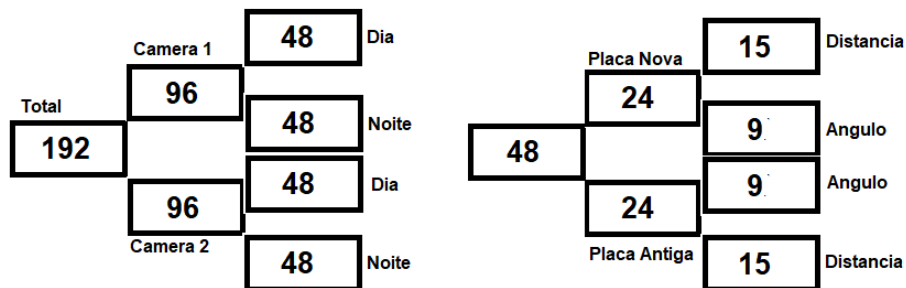


Figura 8: Distribuição dos Métodos de Obtenção de Imagens

Fonte: Autoria Própria

Todas as imagens foram submetidas a fatores externos tais como luminosidade e reflexo do sol durante o dia, assim como iluminação noturna de vias e de estabelecimentos próximos para o caso noturno. Estes fatores podem influenciar no fator de reflexão das placas, dificultando sua localização pelo algoritmo para recorte assim como posteriormente para o reconhecimento dos caracteres.

A fotos realizadas utilizando o parâmetro de distância foram tiradas entre 10 e 30 metros, com um intervalo de 5 metros entre elas, como é ilustrado na Figura 9. A distância inicial foi definida baseada na distância dos radares de velocidade fixos, posicionados em rodovias. Os três sensores presentes no asfalto, responsáveis pelo cálculo de velocidade estão dispostos a três metros um do outro, resultando em 9 metros antes de a foto ser tirada caso haja infração (GALILEU, 2019).

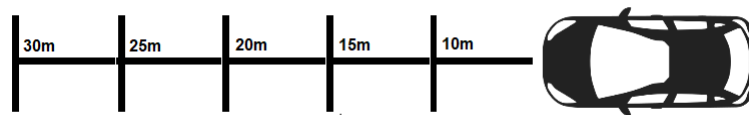


Figura 9: Distância das Fotos

Fonte: (ICONS, 2019) (Adaptado)

Já as fotos com diferentes ângulos foram obtidas a uma distância de 10 metros somente de um lado entre os ângulos de 90 e 30 graus, como é ilustrado na Figura 10. Para cada distância ou ângulo três fotografias foram tiradas, a fim de analisar caso houvesse diferença, 3 fotos para 10 metros, 3 para 15 metros, 3 para 90 graus, 3 para 50 e assim consecutivamente.

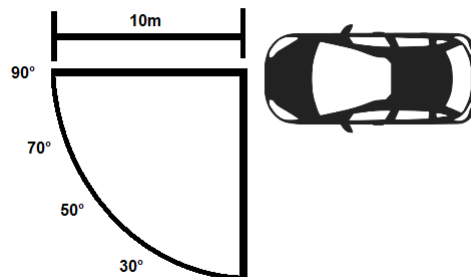


Figura 10: Ângulos das Fotos

Fonte: (ICONS, 2019) (Adaptado)

A segunda etapa da aquisição de dados foi através da gravação de dois vídeos em ambiente controlado utilizando uma distância aproximada de 15 metros em um ângulo de 90 graus, os mesmos submetidos ao processo completo do algoritmo, sendo este a identificação do *frame* do vídeo contendo o carro, recorte da área da placa, segmentação dos caracteres da imagem e reconhecimento dos mesmos.

5.2 PROCESSAMENTO DO ALGORITMO DE RECONHECIMENTO

Na primeira etapa de experimentação, o algoritmo de reconhecimento foi testado, onde o objetivo era a classificação entre placa antiga ou placa Mercosul, as três imagens de cada fator de teste foram processadas e passaram pelo processo de transformação para tons de cinza, localização da placa, recorte, identificação de cor e validação qualidade da imagem para diferenciação entre Placa Nova ou Placa Antiga. Podemos observar os passos na Figura 11 para demonstrar caso a placa seja antiga e na Figura 12, caso seja o formato Mercosul.

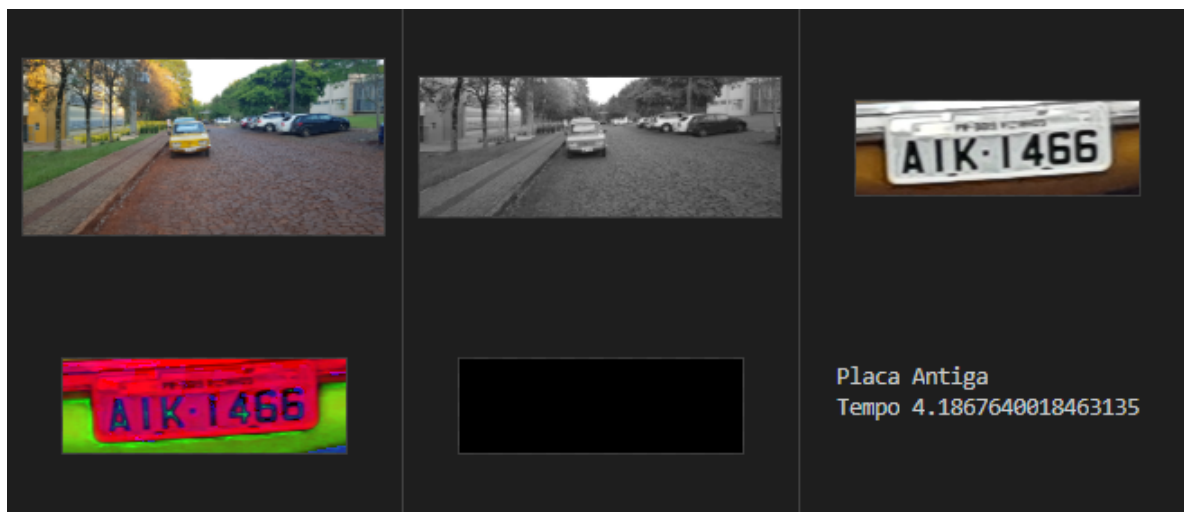


Figura 11: Demonstração Placa Antiga

Fonte: Autoria Própria

Observou-se que, caso o algoritmo não consiga realizar a localização e o corte da região onde está a placa, ele não conseguirá identificar qual o tipo da placa, fato que aconteceu em algumas imagens por estarem em uma distância maior e/ou com ângulo muito acentuado. Caso haja um corte, mas a placa não esteja contida no mesmo ele assume ser uma imagem que contém uma placa antiga. Chegou a ocorrer também que algumas placas Mercosul foram encontradas, no entanto os tons de azul não foram suficientes para identificá-la, assumindo assim a mesma como formato antigo.

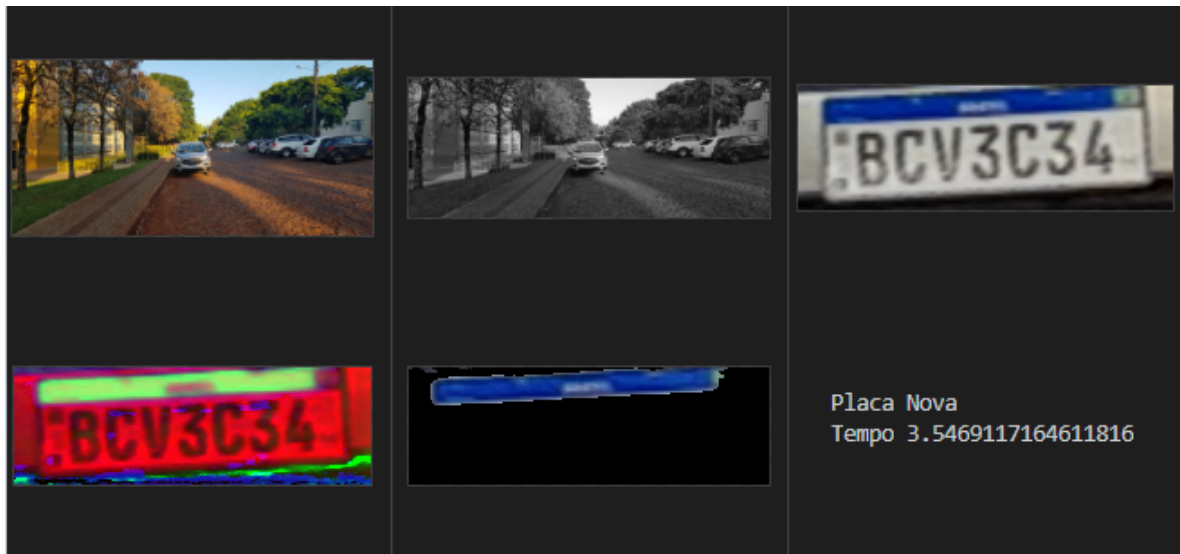


Figura 12: Demonstração Placa Mercosul

Fonte: Autoria Própria

Na experimentação utilizando os vídeos, foi necessária a detecção do *frame* contendo o veículo, para isso foi utilizado o fator de movimento, assim como o tamanho do mesmo. Dessa forma a aplicação da distância adequada foi muito relevante, assim quando um determinado contorno cruza uma linha inserida, um gatilho é acionado retirando uma imagem como podemos ver na Figura 13. Podemos ver o acionamento do gatilho pois a cor da linha é alterada. Após detectada a imagem, esta é encaminhada para o recorte da placa realizando o mesmo processo aplicado na Figura 11 e na Figura 12.



Figura 13: Demonstração da Detecção do Veículo

Fonte: Autoria Própria

A imagem contendo somente a placa é então submetida a uma série de processamentos: a imagem é normalizada (*i.e.*, seus *pixels* são melhor distribuídos), redimensionada para uma escala maior, e sofre um processo de binarização (ANTONELLO, 2014). Assim a imagem recortada é submetida para a segmentação dos caracteres. Todo o processo é demonstrado na Figura 14.

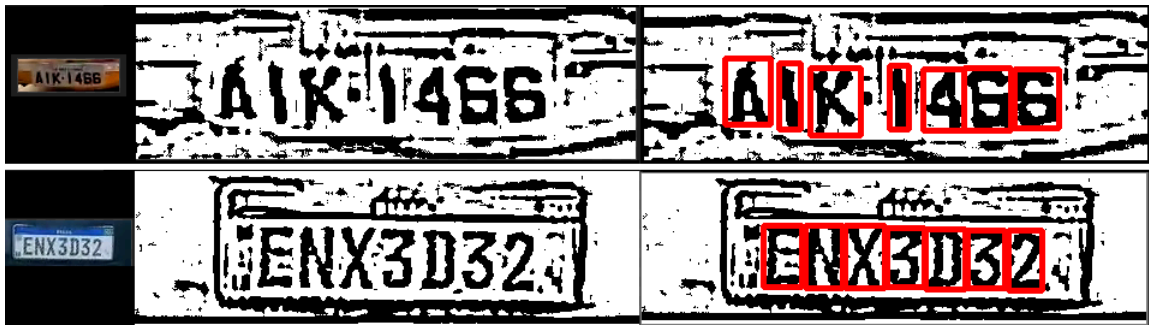


Figura 14: Demonstração das etapas de processamento da imagem da placa

Fonte: Autoria Própria

Para a segmentação, partindo do princípio que sabemos as proporções da imagem, os caracteres irão possuir uma área e tamanhos específicos, assim aplicando uma lógica para extração de contornos e área dos mesmos, é possível encontrar os caracteres. Quando segmentados, e sabendo então a posição e se o mesmo é número ou letra pelo tipo da placa, individualmente cada caractere é submetido ao processamento de uma rede neural convolucional. Foram criadas duas redes neurais, uma para reconhecimento de números de 0 a 9 e a segunda para letras de A a Z.

5.3 RESULTADOS FINAIS

Nesta seção são apresentados os resultados obtidos na ordem do desenvolvimento, onde no primeiro instante foi realizado somente o teste e classificações dos recortes das áreas e tipos das placas, organizados como descrito nas tabelas, segundo os critérios de estudo citados na Seção 4.3, e em um segundo instante o processo completo englobando a detecção do veículo, corte da placa, reconhecimento do tipo, segmentação dos caracteres e reconhecimento dos mesmos.

Todas as 192 imagens fotografadas foram processadas e avaliadas para isso montou-se tabelas, as quais possuem o fator de estudo na primeira coluna, o resultado de corte e reconhecimento entre nova e antiga e o tempo do processamento do algoritmo em segundos, para os dois tipos de placas. Ressalta-se que algumas imagens foram reutilizadas.

Do total de 192 imagens, 24 foram reprocessadas: as imagens tiradas a 10 metros de distância em seus respectivos fatores foram reutilizadas para representar o ângulo de 90 graus, assim os resultados são apresentados contabilizando 216 imagens.

Foram geradas oito tabelas com os dados que são listadas a seguir, os números representados por $X/3$ significam os acertos do total de 3 fotos para cada fator de estudo. Todos os fatores de estudo foram tabulados, são eles: Dois tipos de placas; Resolução de câmeras diferentes; Fator luminosidade (dia e noite); Fator Distância da placa; Fator ângulo da placa.

Um total de 114 cortes foram realizados com sucesso, 64 no recorde das placas antigas e 50 no recorte de placas novas. No entanto somente 87 desses recortes foram classificados de forma correta, os mesmos 64 para as placas antigas e 23 para placas novas. Pelo fato de o algoritmo muitas vezes não encontrar o tom da placa nova, a placa era classificada como antiga, não contabilizando assim o acerto para a placa nova.

A câmera semi-profissional, por ter uma maior qualidade de imagem, repercutiu em um reconhecimento maior, do total de 201 acertos para recorte e reconhecimento das duas câmeras, 112 foram da câmera 2 e 89 da câmera 1. No quesito dia e noite, a margem de acerto também foi divergente, 129 acertos para o dia contra 72 para a noite.

É evidente que um fator influencia o outro, mas o principal influenciador foi luminosidade, durante a noite, muitas placas não foram encontradas nas imagens, assim como houve uma maior dificuldade em encontrar o tom de azul da placa nova. A distância da fotografia é um fator que deve ser estudado com cuidado: observou-se que quanto mais próximo, maior a taxa de acerto. Nota-se uma maior dificuldade no processamento das placas no padrão Mercosul, principalmente nas fotos tiradas durante a noite. Durante a realização das fotografias, o olho humano já encontrava dificuldade de detectar o azul da placa, assim como para ler os caracteres; tal especulação foi confirmada pelo algoritmo.

Os resultados obtidos quanto ao tempo de processamento do algoritmo foram que: O tempo das imagens fotografadas com a segunda câmera foi maior que as fotografias da câmera 1, devido a qualidade da imagem, pois quanto mais *pixels* tiver a imagem mais custoso será o mapeamento para encontrar, recordar e detectar a cor da placa, no entanto como exposto anteriormente a mesma lucrou maior sucesso nessa tarefa. A Figura 15 apresenta gráficos de comparação para o tempo em segundos de processamento do algoritmo para cada câmera, separados para placa nova e antiga, o eixo y representa o tempo e o eixo x representa cada imagem inserida, a cor azul representa uma imagem da câmera 1 e o alaranjado a da câmera 2;

Na Tabela 1 e Tabela 2 são apresentados os resultados decorrentes da experimentação comparando placa antiga e nova, utilizando o fator distância e resolução de imagens capturadas durante o dia com a câmera 1 e câmera 2 respectivamente.

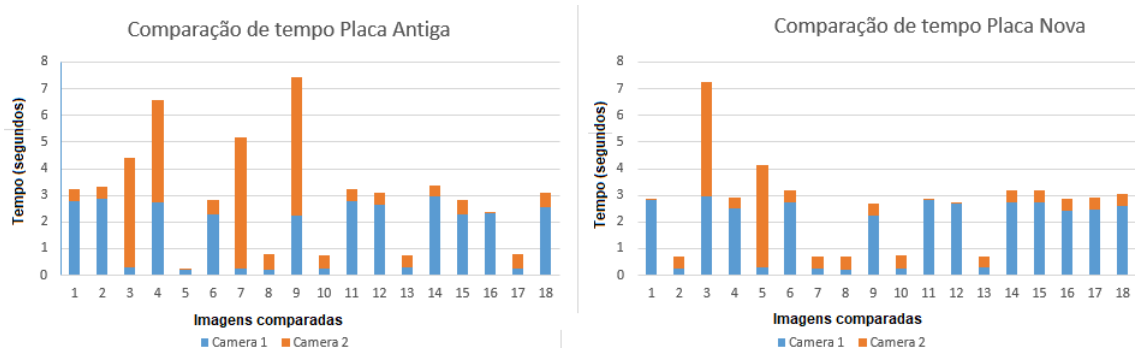


Figura 15: Comparação de tempo Placa Antiga e Placa Nova

Fonte: Autoria Própria

Tabela 1: Taxas de Acerto com fatores (Distância, Câmera 1, Dia)

Fator de estudo: Distância e resolução da imagem.						
Fotos tiradas durante o dia com câmera 1.						
	Placa Antiga			Placa Nova		
Distância	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
10m	2/3	2/3	2,78s	2/3	1/3	2,82s
15m	3/3	3/3	2,86s	1/3	1/3	2,50s
20m	3/3	3/3	2,94s	0/3	0/3	2,98s
25m	3/3	3/3	2,72s	0/3	0/3	2,51s
30m	1/3	1/3	1,99s	0/3	0/3	2,80s
Total	12/15	12/15	2,72s	3/15	2/15	2,51s

Na Tabela 3 e Tabela 4 são apresentados os resultados decorrentes da experimentação comparando placa antiga e nova, utilizando o fator distância e resolução de imagens capturadas durante a noite com a câmera 1 e câmera 2 respectivamente.

Na Tabela 5 e Tabela 6 são apresentados os resultados decorrentes da experimentação comparando placa antiga e nova, utilizando o fator ângulo e resolução de imagens capturadas durante o dia com a câmera 1 e câmera 2 respectivamente.

Na Tabela 7 e Tabela 8 são apresentados os resultados decorrentes da experimentação comparando placa antiga e nova, utilizando o fator ângulo e resolução de imagens capturadas durante a noite com a câmera 1 e câmera 2 respectivamente.

Em um segundo momento utilizando toda a aplicação foram conduzidos alguns

Tabela 2: Taxas de Acerto com fatores (Distância, Câmera 2, Dia)

Fator de estudo: Distância e resolução da imagem.						
Fotos tiradas durante o dia com câmera 2.						
Placa Antiga				Placa Nova		
Distância	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
10m	3/3	3/3	4,27s	3/3	3/3	4,64s
15m	2/3	2/3	4,53s	1/3	1/3	4,51s
20m	3/3	3/3	4,09s	2/3	1/3	4,26s
25m	0/3	0/3	3,84s	0/3	0/3	4,15s
30m	0/3	0/3	3,81s	0/3	0/3	3,83s
Total	8/15	8/15	4,53s	6/15	5/15	4,51s

Tabela 3: Taxas de Acerto com fatores (Distância, Câmera 1, Noite)

Fator de estudo: Distância e resolução da imagem.						
Fotos tiradas durante a noite com câmera 1.						
Placa Antiga				Placa Nova		
Distância	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
10m	3/3	3/3	2,28s	1/3	0/3	2,71s
15m	2/3	2/3	2,63s	0/3	0/3	2,41s
20m	0/3	0/3	2,21s	0/3	0/3	2,26s
25m	0/3	0/3	2,21s	0/3	0/3	2,22s
30m	0/3	0/3	2,36s	0/3	0/3	2,46s
Total	5/15	5/15	2,63s	1/15	0/15	2,46s

Tabela 4: Taxas de Acerto com fatores (Distância, Câmera 2, Noite)

Fator de estudo: Distância e resolução da imagem.						
Fotos tiradas durante a noite com câmera 2.						
Placa Antiga				Placa Nova		
Distância	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
10m	2/3	2/3	5,17s	2/3	0/3	4,83s
15m	2/3	2/3	4,93s	2/3	0/3	4,73s
20m	3/3	3/3	5,73s	0/3	0/3	4,90s
25m	0/3	0/3	5,23s	0/3	0/3	4,80s
30m	0/3	0/3	4,93s	0/3	0/3	5,06s
Total	7/15	7/15	5,73s	4/15	0/15	4,83s

Tabela 5: Taxas de Acerto com fatores (Ângulo, Câmera 1, Dia)

Fator de estudo: Ângulo e resolução da imagem.						
Fotos tiradas durante o dia com câmera 1.						
	Placa Antiga			Placa Nova		
Ângulo	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
90°	2/3	2/3	2,78s	2/3	1/3	2,82s
70°	3/3	3/3	2,64s	3/3	2/3	2,68s
50°	3/3	3/3	2,83s	3/3	2/3	2,74s
30°	0/3	0/3	2,94s	1/3	1/3	2,75s
Total	8/12	8/12	2,71s	9/12	6/12	2,72s

Tabela 6: Taxas de Acerto com fatores (Ângulo, Câmera 2, Dia)

Fator de estudo: Ângulo e resolução da imagem.						
Fotos tiradas durante o dia com câmera 2.						
	Placa Antiga			Placa Nova		
Ângulo	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
90°	3/3	3/3	4,27s	3/3	3/3	4,64s
70°	3/3	3/3	4,31s	3/3	3/3	4,65s
50°	3/3	3/3	4,62s	1/3	1/3	4,19s
30°	2/3	2/3	4,46s	3/3	2/3	4,27s
Total	11/12	11/12	4,38s	10/12	10/12	2,32s

Tabela 7: Taxas de Acerto com fatores (Ângulo, Câmera 1, Noite)

Fator de estudo: Ângulo e resolução da imagem.						
Fotos tiradas durante a noite com câmera 1.						
	Placa Antiga			Placa Nova		
Ângulo	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
90°	3/3	3/3	2,28s	1/3	0/3	2,71s
70°	3/3	3/3	2,31s	3/3	0/3	2,40s
50°	0/3	0/3	2,31s	2/3	0/3	2,46s
30°	0/3	0/3	2,55s	0/3	0/3	2,59s
Total	6/12	6/12	2,30s	6/12	0/12	2,53s

Tabela 8: Taxas de Acerto com fatores (Ângulo, Câmera 2, Noite)

Fator de estudo: Ângulo e resolução da imagem.						
Fotos tiradas durante a noite com câmera 2.						
	Placa Antiga			Placa Nova		
Ângulo	Corte Imagem	Reconhecido	Tempo	Corte Imagem	Reconhecido	Tempo
90°	2/3	2/3	5,17s	2/3	0/3	4,83s
70°	2/3	2/3	4,83s	3/3	0/3	4,65s
50°	2/3	2/3	5,53s	3/3	0/3	4,55s
30°	1/3	1/3	5,49s	3/3	0/3	4,71s
Total	7/12	7/12	5,33s	11/12	0/12	4,68s

testes utilizando o tensorflow para reconhecimento dos caracteres, para isso foi utilizado o padrão inglês de reconhecimento pois o objetivo era decifrar individualmente os algarismos. Devido a este fator utilizou-se a configuração `config='-psm 10'` a qual reconhece caracteres separadamente. No entanto como as imagens recortadas tem uma qualidade baixa, assim como distorções nos contornos dos caracteres, principalmente na placa Mercosul, por esta possuir marca d'água em todo o corpo, os acertos eram quase inexistentes, sendo assim a utilização desta tecnologia foi descartada.

Como alternativa as duas redes neurais convolucionais foram estruturadas, o treinamento individual destas se deu pela inserção de duas imagens de caracteres, um com fonte referente a da placa antiga e a outra placa nova, por exemplo: Para as letras de A à Z uma imagem com fonte *Mandatory* foi inserida para a placa antiga e uma com a fonte *FE Engelschrift* para a placa nova, o mesmo processo foi aplicado para os números de 0 a 9, as imagens foram extraídas dos sites de cada fonte. Cada uma das redes neurais foi submetida a um treinamento de 5000 épocas, os resultados do treinamento de cada uma são apresentados a seguir.

A RNA de números apresentou uma exatidão de 0.9444% com um tempo de *1s 65ms/step* treinando com um total de 18 imagens, 9 para cada fonte, já a RNA de letras apresentou uma exatidão de 0.9783% com um tempo de *1s 30 ms/step* treinando com um total de 46 imagens, 23 para cada fonte. Observa-se na Figura 16, a matriz de confusão exibindo o resultado do desempenho algoritmo de classificação, na Figura 17 e que os treinamentos aumentaram inversamente os valores de perda, assim como ficou nítido na Figura 18 que a quantidade de épocas de treinamento foi muito grande, pois após aproximadamente 400 épocas os valores de acerto ficaram altos e se mantiveram estáveis, sendo este um dos motivos para a realização de uma segunda rodada de treinamentos.

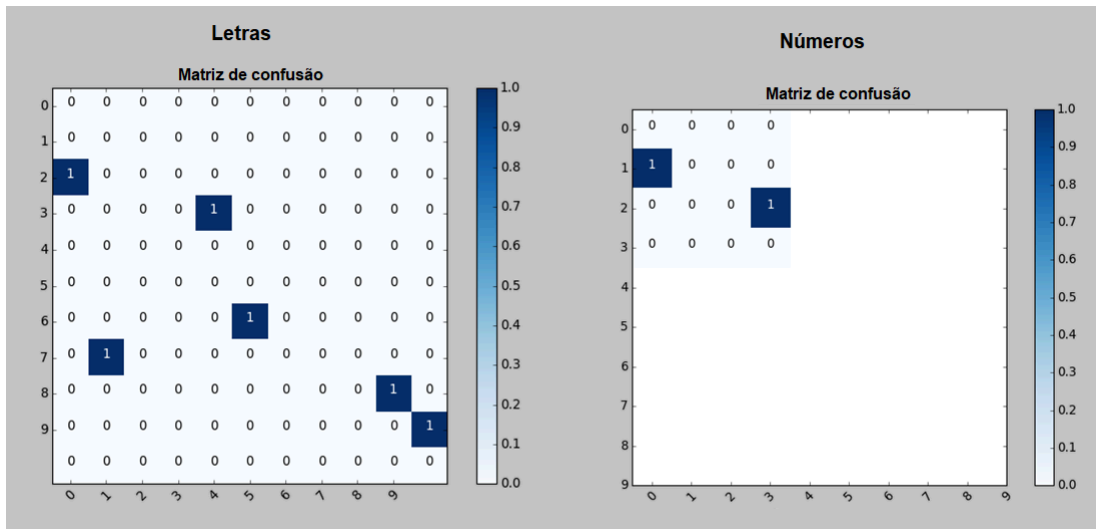


Figura 16: Matrizes

Fonte: Autoria Própria

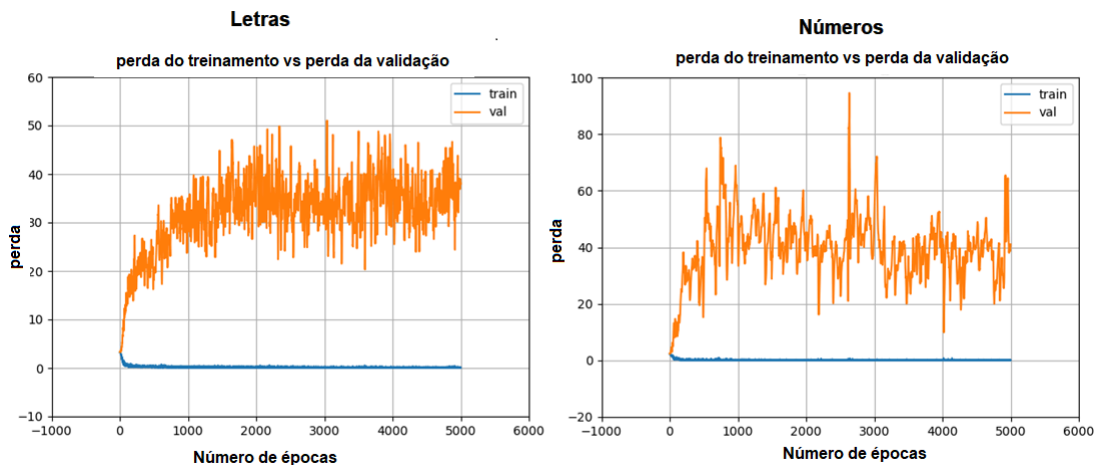


Figura 17: Treinamento 1

Fonte: Autoria Própria

Com o algoritmo finalizando, foi iniciado o processo de teste, identificando previamente o tipo da placa e separado seus caracteres, sabe-se qual a posição e tipo dos mesmos, assim é possível saber qual das RNAs irá receber o segmento da placa para reconhecer.

Após a primeira etapa de treinamento e testes, o algoritmo lucrou assertividade em alguns dos algarismos detectados, no entanto nem todos foram de forma correta, notou-se então a necessidade de expansão da base de treino. Como encontrar caracteres adequados demanda muito tempo, a solução foi “viciar” as RNAs para saber se estas apresentariam os

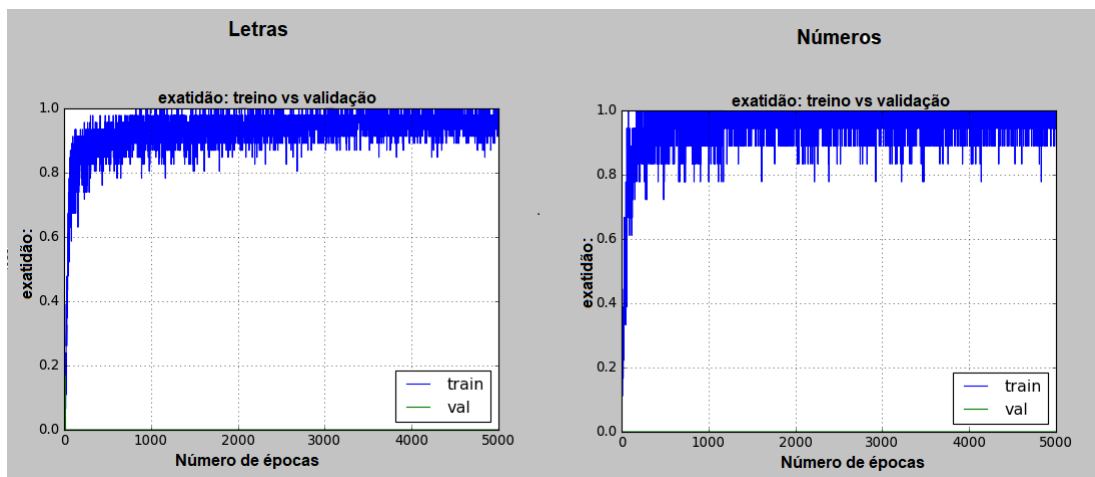


Figura 18: Resultados do Treinamento 1

Fonte: Autoria Própria

resultados esperados. Sendo assim os caracteres das placas conduzidos na experimentação sob ambiente controlado foram extraídos e inseridos cada qual na sua base de treino, e após o ajuste para um treinamento de 200 épocas para números e 2000 épocas para letras os resultados foram os seguintes:

Um total de 158 imagens de números forma utilizadas no treinamento, estas encontradas nas placas, e na própria base de treino utilizada como exemplo pelo Professor André Gustavo Schaeffer. Após uma série de treinos assistidos, o resultado final foi de 100% de reconhecimento numérico para os dois tipos de placa com uma exatidão de 0.9051%. Já para a rede neural de letras um total de 54 imagens foram utilizadas, todas retiradas da placa e de sites cujas fontes das placas eram encontradas, oficializando um reconhecimento de 100% e exatidão de 0.9444%, para os testes aplicados em ambiente simulado.

As comparações dos dados de treinamento podem ser observadas nas Figuras 19, mostrando visivelmente a expansão do treinamento, na 20 comparando os valores de perda e 21, onde pode-se observar os valores de acerto e de treinamento.

Quando todo o processo de detecção, recorte da placa, identificação do tipo, extração e reconhecimento dos caracteres é finalizada, o resultado obtido juntamente com o tipo da placa, é um texto com os algarismos da placa, o qual pode ser usado em várias soluções como descrito na Introdução deste trabalho. Para exemplo este trabalho aplicou uma simulação de controle de estacionamento.

A aplicação responsável por gerir de forma básica um estacionamento, foi então integrada com o algoritmo de reconhecimento. Quando o *output* deste é fornecido,

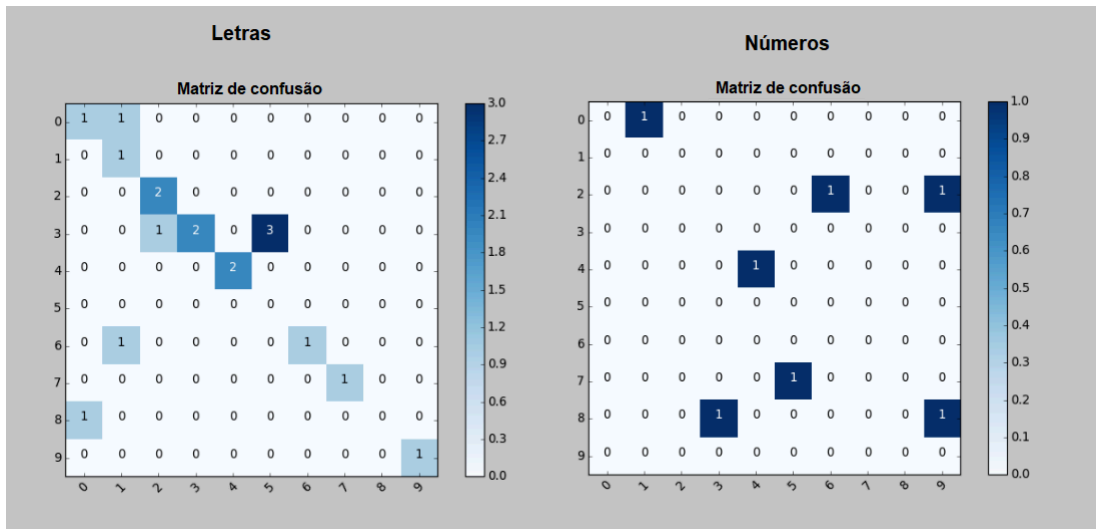


Figura 19: Matrizes Treinamento 2

Fonte: Autoria Própria

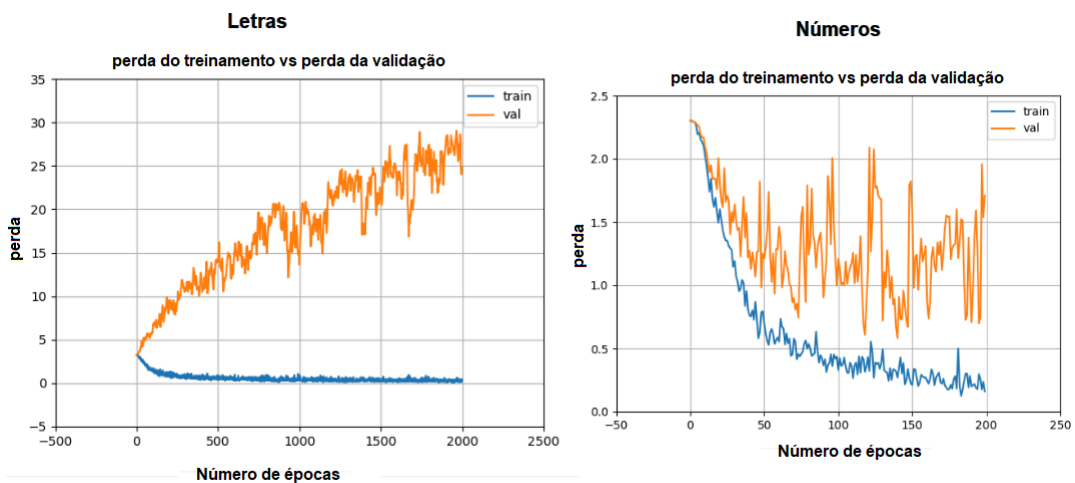


Figura 20: Treinamento 2

Fonte: Autoria Própria

uma requisição é disparada ao *back-end*, que consiste em uma *Application Programming Interface Representational State Transfer* (API REST), desenvolvida utilizando o padrão de desenvolvimento *Model-View-Controller* (MVC), assim que o texto da placa é recebido existe uma validação de permissão em uma base de veículos pré cadastrados, o resultado deste é disparado através de uma comunicação bidirecional (*Socket.io*) para a aplicação Web chama de de System Plate Control, mostrando ao usuário que um veículo foi identificado e sua respectiva permissão.

O *back-end* da aplicação se comunica com um banco de dados não relacional e com

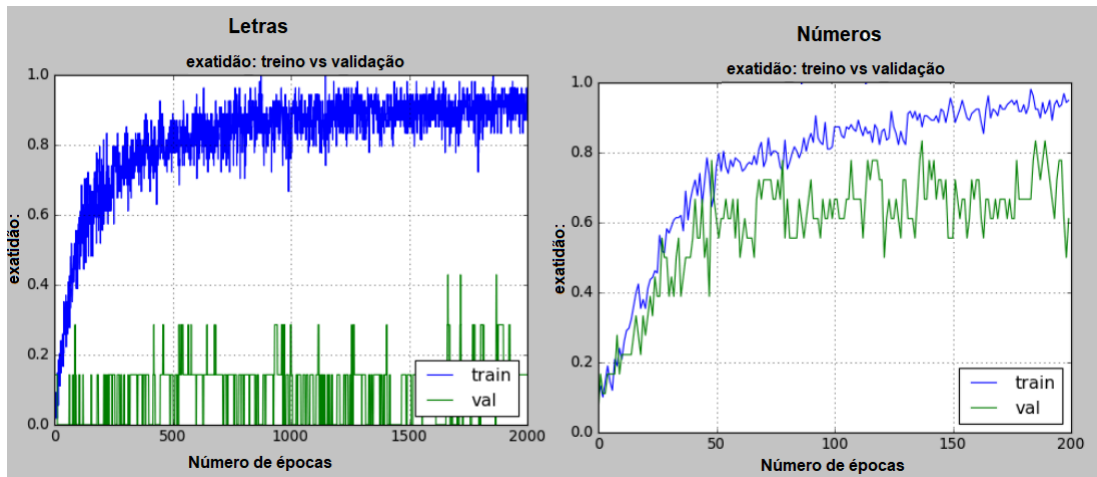


Figura 21: Resultados do Treinamento 2

Fonte: Autoria Própria

o *front-end* desenvolvido, composto por 8 telas, sendo estas: telas de *login* e cadastro, tela de indicadores do estacionamento (início), exibindo o número de vagas totais e a quantidade disponível, tela de perfil que contém os dados do usuário. Dentre todas as principais telas são a de cadastro e listagem de veículos vistos nas Figuras 22 e 23, responsáveis por adicionar ao banco de dados os veículos com permissão de acesso ao estacionamento.

The screenshot shows a web browser window displaying a web application. The browser's address bar shows 'localhost:4200/home'. The application has a blue header with the text 'Plate Control' and a user profile icon. Below the header, there is a section titled 'Cadastro de Veículos' with a toggle switch for 'Placa Mercosul'. The form contains several input fields: 'Modelo *', 'Cor *', and 'Ano *' (with a dropdown arrow), and 'Número da Placa: AAA-0000'. At the bottom of the form, there are two buttons: 'Voltar' (highlighted with a red border) and 'Salvar' (highlighted with a green border). The Windows taskbar is visible at the bottom of the screen, showing the date and time as 18:32 on 27/10/2020.

Figura 22: Tela de cadastro dos veículos

Fonte: Autoria Própria

As outras duas telas também mais relevantes ao sistema, são a de cadastro de estacionamento, responsável por indicar a quantidade de vagas do mesmo e o possível

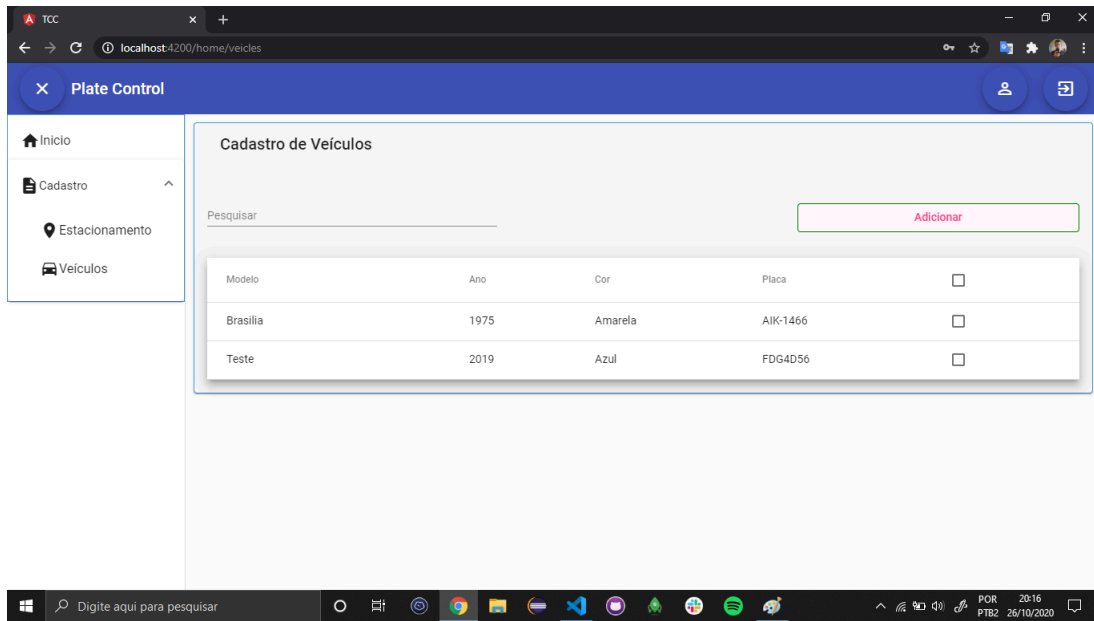


Figura 23: Tela de listagem dos veículos

Fonte: Autoria Própria

custo de uma vaga, visto na Figura 24, e a de veículos que possuem permissão e estão no momento no estacionamento, esta apresentada na Figura 24.

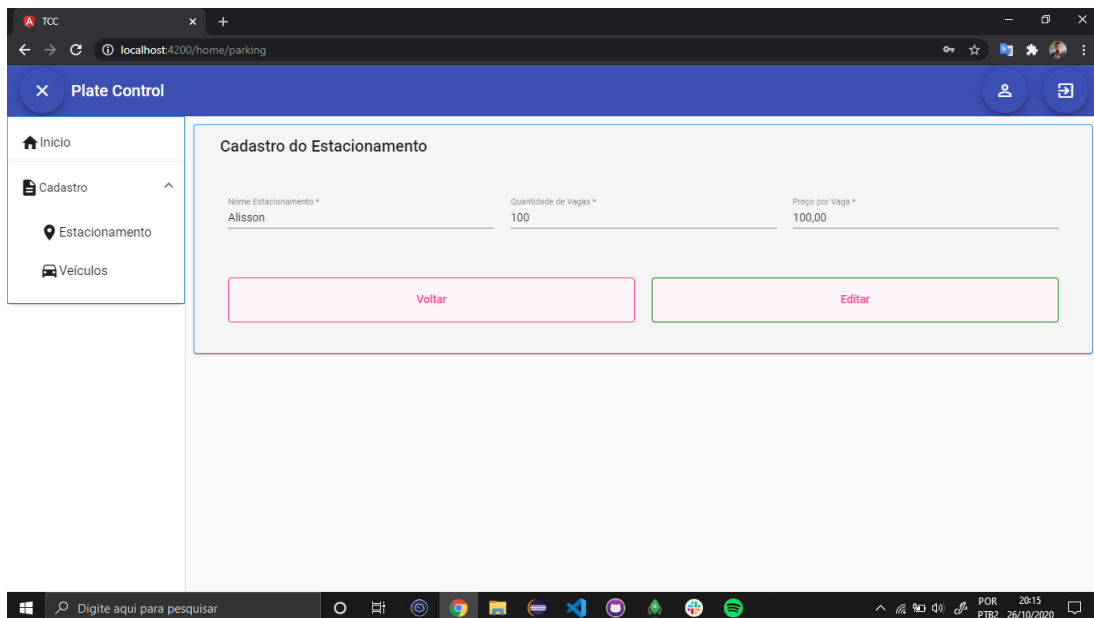


Figura 24: Tela de estacionamento

Fonte: Autoria Própria

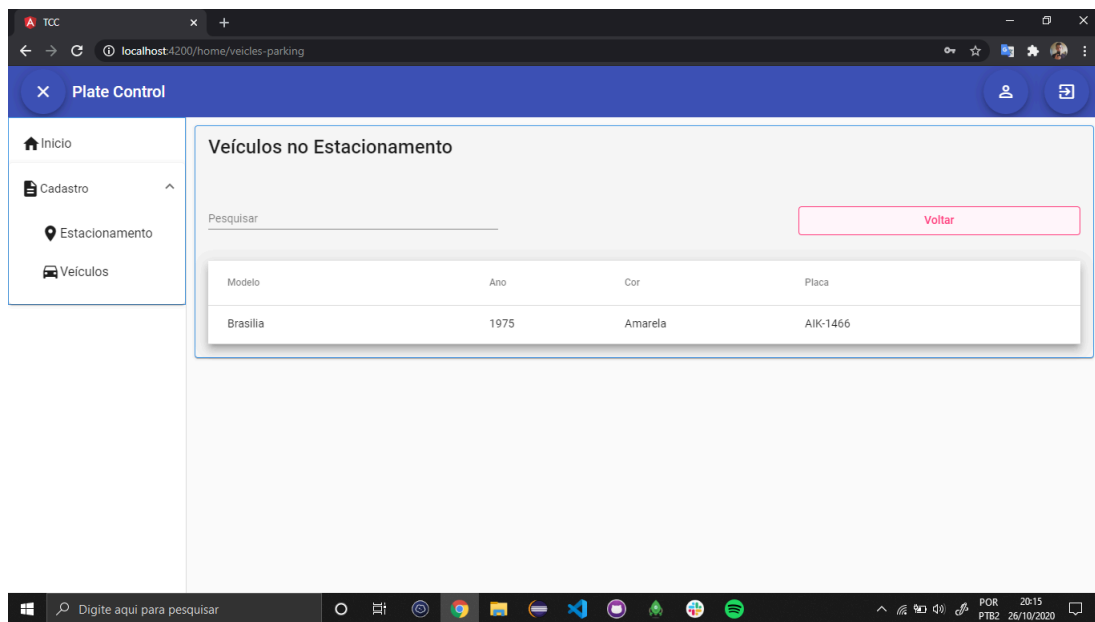


Figura 25: Tela de veículos no estacionamento

Fonte: Autoria Própria

6 CONSIDERAÇÕES FINAIS

A classificação entre placa nova e placa antiga, além de permitir os estudos das diferenças entre as mesmas, contribui para uma maior facilidade no posterior reconhecimento de caracteres, pois é possível diferenciar qual é número e qual é letra através das posições dos caracteres segmentados. Sabemos que a placa antiga tem o formato (*e.g.*, AAA-XXXX) e a placa nova tem o formato (*e.g.*, AAAXAXX), onde “A” representa letra e “X” número, se os caracteres forem segmentados e armazenados em um vetor juntamente com o tipo da placa, podemos auxiliar a Rede Neural no processamento.

Existe precariedade em relação a estudo de placas de motocicletas, durante a elaboração deste trabalho, não foram encontrados estudos sobre. Deixa-se em aberto a questão para estudos futuros, e/ou melhoria deste estudo para poder reconhecer placas de motos. Como o treinamento para o recorte das placas foi obtido de outro trabalho, foi testado também o recorte para placas de motos, não se obtendo sucesso no procedimento, pois a placa não era encontrada.

O estudo foi extremamente satisfatório de ser desenvolvido, e apesar de um longo percurso, algumas melhorias podem ser aplicadas, como por exemplo a expansão da base de teste, melhoria do sistema Web, otimização do tempo de processamento do algoritmo são os principais. Além de integração com dispositivos de *hardware*, para a aplicação em ambientes reais.

Para que melhorias possam ser realizadas futuramente, além da continuidade do estudo, sendo um a expansão do sistema Web desenvolvido e a implantação do algoritmo desenvolvido em um *Hardware* capaz de monitorar veículos em tempo real. Um possível caso de teste não abordado e aberto para aplicação futura, foi a possibilidade de a câmera responsável pela fotografia do veículo não ser capaz de tirar fotos coloridas, impossibilitando o reconhecimento de cores e posteriormente a classificação da placa entre nova e antiga pelo algoritmo.

Um repositório público foi criado no *GitHub*, onde todo os códigos fontes, estudos

e conteúdos desenvolvidos estão disponíveis para acesso pelo *link* <https://github.com/AlissonStopassole/system-plate-control>.

REFERÊNCIAS

- ACADEMY, D. S. **O QUE É VISÃO COMPUTACIONAL?** 2019. Disponível em: <<http://datascienceacademy.com.br/blog/o-que-e-visao-computacional/>>.
- ANTONELLO, R. Introdução a visão computacional com python e opencv. **Engenharia de Controle e Automação do Instituto Federal Catarinense–IFC, campus Luzerna**, 2014.
- BACKES, A. R.; JUNIOR, J. J. d. M. S. **Introdução à visão computacional usando Matlab**. [S.l.]: Alta Books Editora, 2019.
- BOOK, D. L. **As 10 Principais Arquiteturas de Redes Neurais**. 2019. Disponível em: <<http://deeplearningbook.com.br/as-10-principais-arquiteturas-de-redes-neurais/>>.
- BUSSADOR, A. **Localização Automática de Placas de Veículos em Fotos Digitais Utilizando Abordagem Granulométrica**. Tese (Doutorado) — Pontifícia Universidade Católica do Paraná, 2009.
- CHACON, G. et al. Aplicação da técnica de momentos invariantes no reconhecimento de padrões em imagens digitais. **NOTAS TÉCNICAS**, v. 1, n. 2, 2011.
- CHODOROW, K. **MongoDB: the definitive guide: powerful and scalable data storage**. [S.l.]: "O'Reilly Media, Inc.", 2013.
- COIMBRA, U. D. **Introdução à Inteligência Artificial**. 2004.
- CONCI, A.; MONTEIRO, L. H. Reconhecimento de placas de veículos por imagem. **UFF, RJ**, 2004.
- EDMONDS, A. R. **Angular momentum in quantum mechanics**. [S.l.]: Princeton university press, 1996.
- FERREIRA, L. S.; SOARES, L. Reconhecimento automatico de placas veiculares. In: **2012 Workshop de Visao Computacional**. [S.l.: s.n.], 2012.
- FONSECA-GALINDO, J. C.; CASTRO, C. L. de; BRAGA, A. de P. Sistema automatico para reconhecimento de placas de autom oveis baseado em processamento digital de imagens e redes perceptron de multiplas camadas. 2016.
- FONTANA, D. R.; MARIM, L. R. Reconhecimento de caracteres através de redes neurais artificiais com aplicação a placas de automóveis. 2010.
- FORMATO, P. **LPR – License Plate Recognition Brazil**. 2019. Disponível em: <<https://www.dobitaobyte.com.br/lpr-license-plate-recognition-brazil-parte-1/>>.
- GALILEU, R. **Como funcionam os radares fixos**. 2019. Disponível em: <<http://revistagalileu.globo.com/Galileu/0,6993,ECT816469-1716-2,00.html>>.

GUINGO, B. C.; RODRIGUES, R. J.; THOMÉ, A. C. G. Reconhecimento automático de placas de veículos automotores através de redes neurais artificiais. In: **2º Congresso Brasileiro de Computação, Itajaí-SC**. [S.l.: s.n.], 2002.

ICONS, F. **Momentos invariantes de uma imagem**. 2019. Disponível em: <<https://icon-icons.com>>.

LAXMI, V.; ROHIL, H. License plate recognition system using back propagation neural network. **International Journal of Computer Applications**, v. 99, n. 8, p. 29–37, 2014.

MENEZES, N. **Introdução à programação com Python: algoritmos e lógica de programação para iniciantes**. [S.l.]: Novatec, 2010.

MERCOSUL. **Países do MERCOSUL**. 2019. Disponível em: <<https://www.mercosur.int/pt-br/quem-somos/paises-do-mercosul/>>.

MITCHELL, T. **Machine Learning**. [S.l.]: McGraw-Hill Book Company, 1997.

NETO, E. C.; FILHO, P. P. R.; NORTE, A. C. Técnicas de pdi e inteligência artificial aplicadas ao reconhecimento de placas de carro nos padroes brasileiros. 2013.

OpenCV. **Classificador em cascata**. 2019. Disponível em: <https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html/>.

OpenCV. **Conversões de espaço de cores**. 2019. Disponível em: <https://docs.opencv.org/master/d8/d01/group_imgproc_color_conversions.html/>.

OpenCV. **Equalização do Histograma**. 2019. Disponível em: <https://docs.opencv.org/3.4/d5/daf/tutorial_py_histogram_equalization.html/>.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PLACA, P. **Placas Mercosul**. 2019. Disponível em: <<https://br.pinterest.com/>>.

PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional**. [S.l.]: "São Paulo: AMGH," , 2011.

RAUBER, T. W. Redes neurais artificiais. **Universidade Federal do Espírito Santo**, 2005.

SILVA, M. S. **JavaScript-Guia do Programador: Guia completo das funcionalidades de linguagem JavaScript**. [S.l.]: Novatec Editora, 2010.

TEIXEIRA, J. **O que é inteligência artificial**. [S.l.]: E-Galáxia, 2019.

TEIXEIRA, P. **Professional Node.js: Building Javascript based scalable software**. [S.l.]: John Wiley & Sons, 2012.

VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In: **Proceedings of the xxix conference on graphics, patterns and images**. [S.l.: s.n.], 2016. v. 1, n. 4.

VINHAES, M. L. **Sistema de Provas Online com desenvolvimento em Python com Django.** 2013. Disponível em: <<http://bsi.uniriotec.br/tcc/textos/2013Vinhaes.pdf>>. Acesso em: 14 de maio de 2019.

ANEXO A – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Prezado(a),

Esta pesquisa visa desenvolver um reconhecimento automático de placas, utilizando-se do mesmo como uma ferramenta para gerir estacionamentos. Esta pesquisa está sendo desenvolvida por Alisson Stopassole, aluno do Curso de Bacharelado em Engenharia de Software da Universidade Tecnológica Federal do Paraná – Campus Dois Vizinhos (UTFPR-DV), sob a orientação do Prof. Dr. Yuri Kaszubowski Lopes e coorientação do Prof. Dr. Gustavo Jansen de Souza Santos.

O objetivo específico do estudo é identificar os possíveis impactos do novo padrão de placas adotado pelo Brasil, o formato Mercosul, através de um reconhecimento automático de placas, pois tais mudanças precisam ser acompanhadas para propor uma solução capaz de automatizar, assim como no formato antigo, o processo de reconhecimento. E utilizar o algoritmo de reconhecimento como uma ferramenta capaz de controlar um estacionamento.

Solicitamos a sua colaboração para com o experimento, por meio do empréstimo do veículo para a realização de fotografias da placa, bem como sua autorização para apresentar os resultados deste estudo em eventos da área de computação e tecnologia e publicar em revista científica nacional e/ou internacional. Por ocasião da publicação dos resultados, seu nome será mantido em sigilo absoluto. Informamos que os procedimentos de coleta não são invasivos e não haverá risco para a sua integridade física e/ou emocional.

Esclarecemos que sua participação no estudo é voluntária e, portanto, o(a) estudante não é obrigado(a) a fornecer as informações e/ou colaborar com as atividades solicitadas pelos Pesquisadores. Caso decida não participar do estudo, ou resolver a qualquer momento desistir do mesmo, não sofrerá nenhum dano. Os pesquisadores estarão a sua disposição para qualquer esclarecimento que considere necessário em qualquer etapa da pesquisa.

Assinatura do pesquisador responsável.

Considerando, que fui informado (a) dos objetivos e da relevância do estudo proposto, de como será minha participação, dos procedimentos e riscos decorrentes deste estudo, declaro o meu consentimento em participar da pesquisa, como também concordo que os dados obtidos na investigação sejam utilizados para fins científicos (divulgação em eventos e publicações).

Dois Vizinhos, _____ de outubro de 2019.

Assinatura do(a) participante.