

## USO DE INTELIGÊNCIA ARTIFICIAL PARA RECONHECIMENTO DE VAGAS DISPONÍVEIS EM ESTACIONAMENTOS

**João Paulo Pizzetti Réos<sup>1</sup>**

**Anderson Rodrigo Farias<sup>2</sup>**

**Resumo:** O trabalho tem como objetivo a utilização da inteligência artificial para encontrar vagas disponíveis em estacionamentos, através do tratamento de imagens. Visando verificar o estado das vagas no meio das grandes cidades, nas quais a situação com relação ao número de veículos piora a cada dia, tornando a tarefa de encontrar um local disponível cada vez mais difícil. Por meio da inteligência artificial e o processamento de imagens, com o auxílio de câmeras de monitoramento ou câmeras próprias implantadas em locais estratégicos, será feita a identificação a fim de determinar se existem vagas disponíveis espalhadas pelas cidades, mostrando este resultado por meio de uma plataforma visual aos motoristas.

**Palavras-chave:** Inteligência Artificial; Cidades Inteligentes; Gerenciador de Vagas.

### 1 INTRODUÇÃO

A quantidade de veículos nas ruas das grandes cidades vem crescendo significativamente, segundo Departamento Estadual de Trânsito de Santa Catarina (DETRAN/SC) existem mais de 4.850.000 veículos automotores circulando no estado no ano de 2018, sendo que em 2017 havia pouco mais de 4.700.000, refletindo em um aumento de 3% em apenas um ano [1]. Estes números podem ser comparados com o número de habitantes no estado, o qual para cada mil habitantes se tem 690 veículos registrados, resultando numa média de um veículo para cada 1,4 habitantes, o que é um número consideravelmente alto quando se pensa em mobilidade urbana e no que diz respeito ao que deveria trazer facilidades, ao mesmo tempo isto gera dores de cabeça aos motoristas [2].

Segundo dados coletados pelo Plano de Mobilidade Urbana Sustentável (PLAMUS), a capital de Santa Catarina, Florianópolis, fica em primeiro lugar no que diz respeito à utilização de veículos automotores individuais para se deslocar pela cidade, ficando à frente de grandes cidades como São Paulo, Rio de Janeiro e até

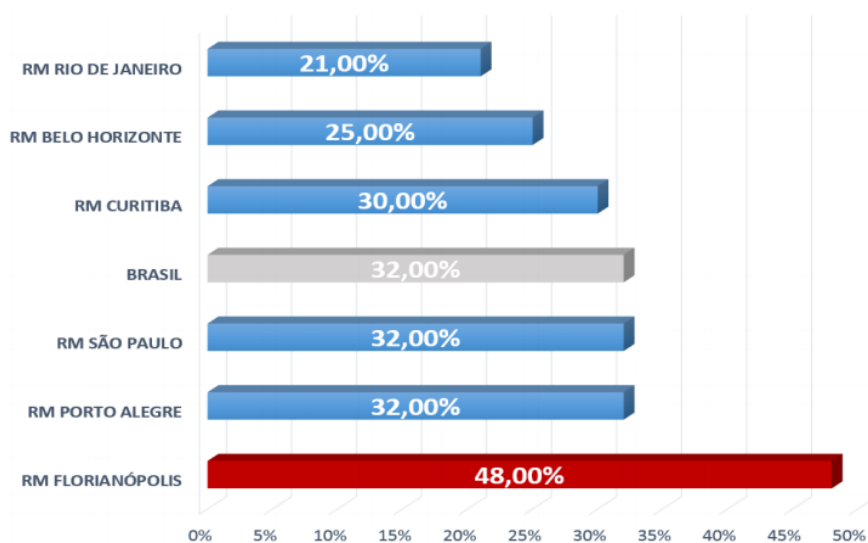
---

<sup>1</sup> Graduando em Engenharia da Computação. E-mail: joao\_paulo\_pizzetti\_reos@hotmail.com

<sup>2</sup> Prof. do curso de Engenharia da Computação. E-mail: anderson.farias@satc.edu.br

mesmo da média geral do Brasil, como pode ser visto na Figura 1 [3]. Dentre os impactos negativos da mobilidade urbana, o congestionamento é o que sempre chamou mais atenção pelos seus prejuízos e pela sua alta visibilidade, causando incômodo às pessoas que vivem nas cidades afetadas, principalmente pelo estresse gerado naqueles que precisam enfrentar estas condições no seu dia a dia [4].

Figura 1 - Percentual de uso de automóveis no total de viagens



Fonte: Baseado em [3].

Diante do significativo aumento de veículos nas grandes cidades, surge uma necessidade que deve ser levada em consideração: encontrar vagas disponíveis para estacionar o seu veículo.

Nos centros das grandes cidades, pode-se encontrar uma variedade imensa de estacionamentos, sejam eles privados ou até mesmo públicos. Na cidade de São Paulo, por exemplo, há à disposição dos motoristas as zonas azuis, que são estacionamentos rotativos regulamentados e pagos, localizados em vias públicas e de fácil acesso aos motoristas. Este tipo de estacionamento contribui para a melhora da mobilidade urbana nas cidades, por meio da rotatividade do uso de vagas, possibilitando que um número maior de veículos as utilize, resultando em menor tempo de procura e, conseqüentemente, melhor fluidez do trânsito [5].

Para auxiliar os motoristas algumas empresas como por exemplo, a Companhia de Engenharia de Trânsito de São Paulo (CET), desenvolveu um

aplicativo que possibilita aos motoristas comprar créditos para estacionar nas zonas azuis, agilizando o processo e acabando com os papéis [5].

Como resultado da evolução das redes convencionais que conectam milhões de dispositivos em todo o mundo, surge a Internet das Coisas (IoT), que tem como conceito conectar estes dispositivos. Com o surgimento de dispositivos inteligentes e os recentes avanços da tecnologia, esta conexão se torna cada vez mais favorável aos olhos dos usuários. A partir do conceito de IoT, surgem as cidades inteligentes, que vêm se tornando o centro das atenções nas últimas décadas, devido aos problemas de urbanização e mobilidade em todo o mundo. Autores afirmam, que as cidades inteligentes são herdeiras do IoT por utilizarem seus componentes essenciais para a sua construção como por exemplo, a geração de dados, o gerenciamento destes dados e o manuseio de aplicativos [6].

Em termos genéricos, cidades inteligentes podem ser conceituadas, como, um ambiente urbanizado que utiliza a Tecnologia da Informação e Comunicação (TIC) e outras tecnologias para melhorar a eficiência e o desempenho de operações regulares fornecidas aos cidadãos. Já em termos formais se define cidades inteligentes dizendo que elas conectam a infraestrutura física, social, empresarial e TIC para elevar a inteligência da cidade [6].

Dessa maneira as cidades inteligentes podem melhorar questões como mobilidade urbana e conseqüentemente a vida dos cidadãos que vivem nelas.

Mesmo com todas as tecnologias existentes no mercado hoje, não deixa de existir um problema que é: onde irá deixar o seu veículo quando há necessidade de realizar alguma função. Além de gerar o estresse do trânsito também causa uma preocupação de deixar o seu veículo em segurança. Para isso acontecer o motorista terá de encontrar uma vaga disponível, em uma cidade lotada, na qual a chance de encontrar um local vago é pequena.

Assim, a proposta desse trabalho é o desenvolvimento de um protótipo de sistema de monitoramento de vagas disponíveis em estacionamentos, no qual se possa mostrar aos motoristas os locais com lugares disponíveis espalhados pela cidade. Isto será feito por meio de uma aplicação visual, antes mesmo de sair de onde está ou até mesmo no caminho para o seu destino, economizando assim o seu tempo de viagem e minimizando as suas preocupações.

## 2 INTELIGÊNCIA ARTIFICIAL

Para o desenvolvimento do protótipo citado, deve-se ter o conhecimento de alguns conceitos importantes, que serão abordados neste capítulo e adotados ao longo do desenvolvimento, como o conceito de inteligência artificial e alguns dos algoritmos presentes no mercado, e o tratamento de imagens, o que pode auxiliar no processo.

### 2.1 CONCEITOS

A inteligência artificial (IA) tem um grande potencial para ajudar a resolver diversos problemas da sociedade, pode-se citar como exemplo, os veículos inteligentes que podem salvar milhares de vidas no mundo inteiro todos os anos. Além disso, aumentar a mobilidade para pessoas incapazes de dirigir um outro tipo de veículo, como deficientes ou idosos. Edifícios inteligentes que podem economizar energia e reduzir a emissão de carbono na atmosfera. Um governo inteligente que possui nas mãos uma forma de servir aos cidadãos de forma rápida e precisa, e também proteger melhor aqueles que se encontram em risco e, como consequência, economizar dinheiro.

Outra área onde é possível aplicar a IA é na educação, de maneira aprimorada, possibilitando aos professores fornecer uma educação que abra portas para uma vida segura e satisfatória. Estas são algumas formas de se aplicar a IA de olho nos benefícios e sempre ficando atento aos riscos e desafios [7].

Mas como se pode definir IA? Para os praticantes não existe uma definição única que possa ser utilizada, pois para alguns IA é um sistema computadorizado que exhibe um comportamento requerente de inteligência, já para outros IA é um sistema capaz de resolver problemas complexos ou tomar ações apropriadas para alcançar objetivos de forma racional [7].

Para os problemas e soluções de IA especialistas oferecem diferentes taxonomias que podem levar em consideração. Dentre elas, há os sistemas que pensam e agem como humanos e os que pensam e agem racionalmente.

Nos sistemas que pensam como humanos, encontram-se a arquitetura cognitiva e as redes neurais. Nos que agem como humanos, existe o processamento

de linguagem natural, com o teste de Turing e a representação de conhecimento. Nos que pensam racionalmente há, em exemplo, as soluções lógicas, inferência e automatização. E, por fim, nos que agem racionalmente, encontram-se os agentes de software inteligentes e robôs que atingem objetivos por meio de percepção, raciocínio, aprendizado, comunicação, tomada de decisão e atuação [8].

No conceito de *Deep Learning* ou aprendizado profundo, que atualmente é a técnica dominante no aprendizado de dados, existe um diversificado conjunto de modelos propostos, os quais podem ser utilizados junto ao tratamento de imagens. Um destes modelos é a *Decision Tree* (DT, em tradução livre: árvore de decisão), comumente utilizada para resolver tarefas de classificação e regressão, representando visualmente e explicitamente as decisões tomadas a partir dos dados inseridos no problema [9]. Este modelo trabalha dividindo o processo em forma de árvore como o próprio nome sugere, realizando uma série de testes consecutivos nos quais cada decisão corresponde a um nó e em cada nó raiz se aplica um teste com o qual o seu resultado dará origem a um nó filho. Quando este atinge uma folha, definição para um nó sem filhos, a decisão atribuída a esta corresponde ao resultado da amostra. O método citado é muito eficiente para algumas ocasiões, como treinamentos e inferência, mas é propenso a ajustes excessivos, nos casos onde o modelo é complexo e se ajusta bem aos dados de treinamento, porém não consegue generalizar para dados não vistos [10].

O algoritmo mais popular quando se fala em árvores de decisão, é o *Iterative Dichotomiser 3* (ID3) desenvolvido por Quinlan. O mesmo possui a ideia básica de construir uma árvore de cima para baixo a partir do nó raiz, onde cada atributo é testado e o que melhor classifica os dados é escolhido, utilizando o ganho de cada atributo para classificar os dados, que por sua vez é calculado a partir da diferença na entropia utilizando Eq. (1) [11]:

$$entropia(D) = - \sum_{i=1}^m p_i \log(p_i) \quad (1)$$

Onde:

D = vetor de observação;

m = número de classe;

$p_i$  = probabilidade de D pertencer a classe i.

A informação do ganho é calculada através da subtração da entropia antes da divisão e após a divisão utilizando Eq. (2), repetindo-se o processo para cada atributo [11]:

$$\text{Ganho}(A) = \text{Info}(D) - \text{Info}_a(D)$$

Onde: (2)

$$\text{Info}_a(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times \text{Info}(D_j)$$

Onde:

A = atributo que está sendo processado;

v = número de valores distintos no atributo A;

$\frac{|D_j|}{|D|}$  = peso da divisão de j.

Ao trabalhar com grandes conjuntos de dados e recursos, onde a utilização de especialistas humanos se torna ineficiente o algoritmo se mostrou eficiente [11].

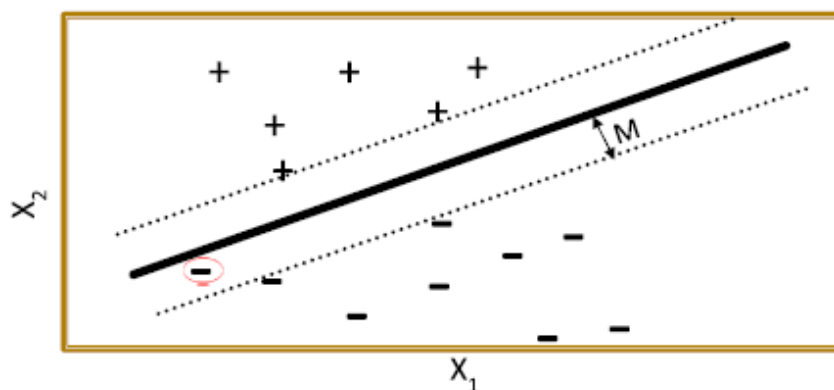
Uma das principais vantagens dos modelos baseados em árvores de decisão, é de que o problema pode ser dividido em vários subproblemas, com regras de decisão claras. Na qual a mesma estratégia pode ser aplicada para resolver todo o problema, o que o torna fácil de ser entendido e implementado. Já por outro lado, a sua maior desvantagem são os ajustes excessivos, citado anteriormente, entretanto podem ser resolvidos definindo restrições nos parâmetros [12].

*Random Forest* (RF em tradução livre: florestas aleatórias) seguem o mesmo modelo das árvores de decisão, utilizadas também para classificação e regressão, uma de suas vantagens por formar a maioria dos sistemas atuais. Uma RF é formada por um conjunto de DTs, que agrupadas formam uma floresta e as torna aleatória. Simplificando, as RFs criam várias DTs e as unem para obter mais precisão e estabilidade, tornando-se assim mais robusta, nos casos onde haja variação de dados [10].

Outro modelo utilizado em diversas aplicações é a *Support Vector Machine* (SVM, em tradução livre: máquina de vetores de suporte). Ela opera em amostras mapeadas em um espaço de representação, criando um separador linear, que faz com que as amostras utilizadas caiam em lados opostos da linha, dependendo de sua

classe. Utilizado para realizar classificações, e diferentemente de classificadores lineares tradicionais, utiliza *kernels* para executar classificações não lineares o que o torna mais eficiente. Especialmente em função do *kernel*, o qual projeta os dados em um espaço de dimensão superior, que em espaço de representação superior uma classificação linear equivale ao mesmo que uma classificação não linear realizada em espaço de representação inferior. Outro fator que torna o SVM mais eficiente que os classificadores tradicionais, é o fato de maximizar as margens entre as classes no hiperplano. No SVM para se determinar a margem máxima utilizada para a classificação, é necessário apenas de um subconjunto dos pontos de entrada, chamados de vetores de suporte, o que é uma de suas grandes vantagens. Na Figura 2 constata-se claramente a definição das margens, delimitadas pelos vetores de suporte (os dois pontos negativos e os dois pontos positivos). Na imagem também há uma desvantagem do SVM, por causa do uso de margens o seu alargamento pode fazer com que pontos apareçam do lado errado e sejam classificados incorretamente [10], [13].

Figura 2 - Delimitação da margem e amostragem do erro, ponto negativo circulado



Fonte: Baseado em [13].

Algumas das vantagens e possibilidades que o SVM proporciona, é o fato de ser eficaz com dimensões elevadas ou onde o número de dimensões é maior que o número de amostras. O seu ponto alto é a utilização de vetores de suporte para delimitação das margens, mas também possui suas desvantagens. O SVM necessita de um tempo de treinamento elevado para grandes conjuntos de dados o que reduz o

seu desempenho, além de existir a possibilidade de ruído nos casos nos quais as classes de destino estão sobrepostas, não trabalhando com toda a sua capacidade [14].

A implementação do método SVM pode ser encontrada em várias aplicações atuais, como previsão de séries temporais, reconhecimento de manuscrito, detecção de rostos, classificação de imagens, entre outras. E ainda deixando em aberto novas possibilidades, como a previsão de vendas, aviso prévio de probabilidade de mudanças para concorrentes ou segmentação de clientes, classificando-os por propensão de compra [13].

## 2.2 TRATAMENTO DE IMAGEM

Com os avanços em *Deep Learning*, são produzidos resultados inovadores com a implementação do tratamento e classificação de imagens. Através disso a utilização de poderosas técnicas de aprendizado de máquina e o crescimento dos recursos gráficos, começam a gerar resultados significativos em várias áreas. Por exemplo como na medicina em seus tratamentos e diagnósticos, na segurança com a implementação de reconhecimento facial na busca de criminosos, em área florestais para busca de desmatamentos ou uso ilegal de áreas verdes e entre outras áreas onde a tecnologia pode auxiliar, substituindo técnicas clássicas [10].

O propósito do aprendizado de máquina é a utilização de um algoritmo no qual se entre com os dados e o processamento seja feito retornando um resultado desejado. E uma das possíveis entradas no algoritmo, são imagens processadas e entendidas com a utilização de computadores que alteram a sua natureza, a partir das texturas e cores. Imagens são formadas a partir de uma junção de pixels, que é a menor unidade que constitui uma imagem digital. Por exemplo, uma simples imagem tirada com uma câmera fotográfica possui em torno de 4 a 6 milhões de pixels, onde cada um corresponde a uma cor em um ponto da imagem [10], [15].

Para a análise de imagens por meio de uma *Application Programming Interface* (API, em tradução livre: interface de programação de aplicações), a utilização de uma biblioteca auxilia muito no momento de criar o programa, e hoje no mercado a mais difundida para este tipo de trabalho é o OpenCV. Esta é uma biblioteca de funções da linguagem C que implementa o processamento de imagens



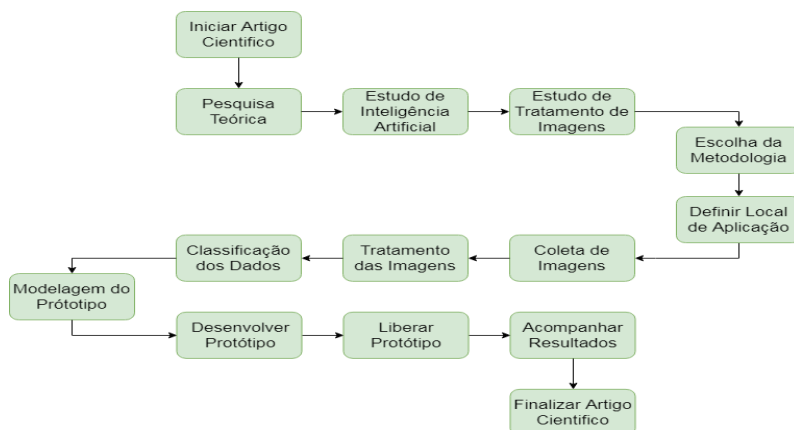
e funções de visão. Como qualquer biblioteca a mesma irá apenas fornecer recursos para o desenvolvimento, tendo o desenvolvedor que criar suas próprias lógicas e funções para alcançar seus objetivos [15].

A utilização do tratamento de imagens vem se tornando cada vez mais popular em situações do cotidiano, como por exemplo no mundo da segurança, tal qual citado anteriormente, com o uso de algoritmos de reconhecimento facial. Foi apresentado em conferência no IEEE o MegaFace, que força estes tipos de algoritmos, como o FaceNet do Google, a realizar verificações e identificações, com o intuito de determinar se duas faces apresentadas ao algoritmo pertencem à mesma pessoa. A identificação envolve encontrar a pessoa em meio a um milhão de distrações. O resultado da implementação no algoritmo do Google, fez com que a precisão quase perfeita caísse para 75%, onde pode se concluir que os algoritmos ainda estão muito aquém da perfeição [16].

### 3 PROCEDIMENTO METODOLÓGICO

Para alcançar os objetivos propostos no projeto, dividiu-se o mesmo em algumas etapas, como demonstrado na Figura 3. A construção da base teórica, e conseqüentemente o referencial bibliográfico, deu-se por meio de pesquisas e publicações, com o objetivo de coletar dados sobre a mobilidade urbana e os conceitos que serão abordados no desenvolvimento do protótipo.

Figura 3 - Fluxo de desenvolvimento do projeto



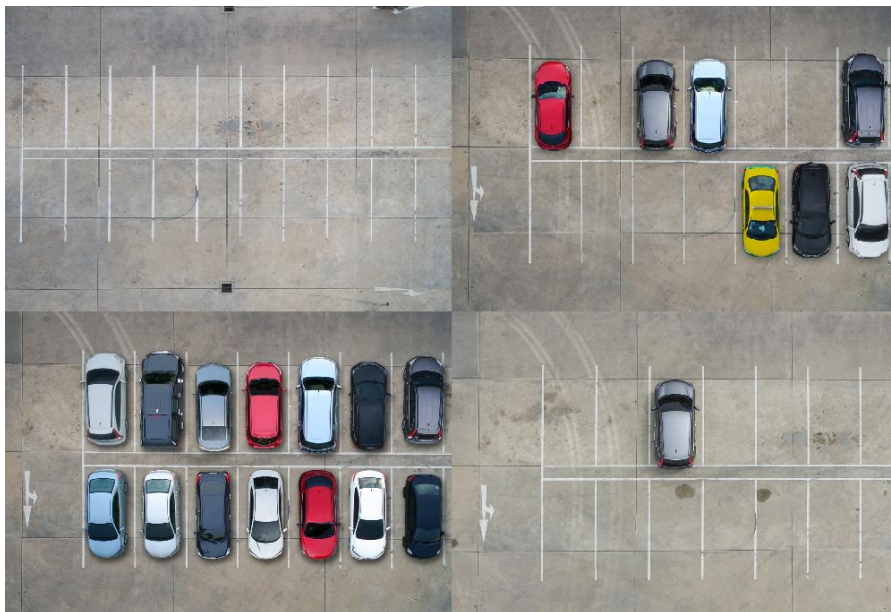
Fonte: do Autor.

Após a conclusão da pesquisa para formação do referencial teórico, foi iniciado um estudo mais profundo nos assuntos relacionados a inteligência artificial e tratamento de imagens, buscando os conceitos necessários para a escolha da metodologia a ser adotada, para assim definir o local de aplicação do protótipo e realizar a coleta das imagens que serão utilizadas para o tratamento e classificação do algoritmo.

### 3.1 TRATAMENTO DOS DADOS

Em um momento inicial o processamento dos dados foi desenvolvido e testado em um ambiente virtual para o treinamento do algoritmo. Para posteriormente ser levada a aplicação para o mundo real e aplicando em estacionamentos. Os dados utilizados no processamento foram encontrados através do Shutterstock [17], no qual há bases de imagens como as representadas na Figura 4.

Figura 4 - Base de imagens aérea de estacionamento



Fonte: modificado e baseado em [17].

Com a base de imagens coletadas se pode dar início ao tratamento dessas imagens, para o reconhecimento das linhas que demarcam cada vaga do

estacionamento e posteriormente os carros que ocupam estas vagas, determinando assim se estas se encontram disponíveis ou não.

Para esta etapa do desenvolvimento foi utilizada a linguagem de programação Python, por se ter um conhecimento prévio sobre o seu funcionamento e pela facilidade do uso desta integrado ao OpenCV, que conforme o capítulo 2.2 da fundamentação teórica, é umas das bibliotecas de desenvolvimento mais difundidas do mercado para este tipo de aplicação. Com o OpenCV vários recursos se tornam disponíveis para o tratamento de imagens e vídeos, onde foram utilizados recursos para converter a imagem recebida para escala de cinza e retirar as informações desnecessárias da imagem.

Após a limpeza da imagem para remoção de informações indesejadas, aplicou-se a função Canny para leitura e retorno das extremidades encontradas na imagem. De acordo com John Canny [18] o algoritmo permite ao desenvolvedor satisfazer três critérios essenciais: a baixa taxa de erros, a boa localização e a resposta mínima. Dentre estes, a mais importante para o desenvolvimento é a baixa taxa de erros, por ter uma boa detecção apenas das arestas existentes e a resposta mínima, por retornar apenas uma resposta da detecção por aresta encontrada na imagem recebida.

A partir desta etapa, realizou-se o reconhecimento de cada vaga do estacionamento, para posteriormente identificar o status de disponível ou ocupado. Para isso se utilizou a transformada de Hough, encontrada em diversos projetos, tendo como objetivo o reconhecimento de linhas em imagens distintas, isto pelo fato de reduzir consideravelmente o ruído das imagens quando comparado a outros métodos disponíveis no mercado [19]. Por isso também se justifica a utilização da função Canny citada anteriormente, na qual as extremidades presentes na imagem são reconhecidas e com o seu resultado obtido se submete este a transformada que irá se encarregar de reconhecer as linhas, e desta forma se chega ao reconhecimento das vagas, objetivo desta etapa do desenvolvimento.

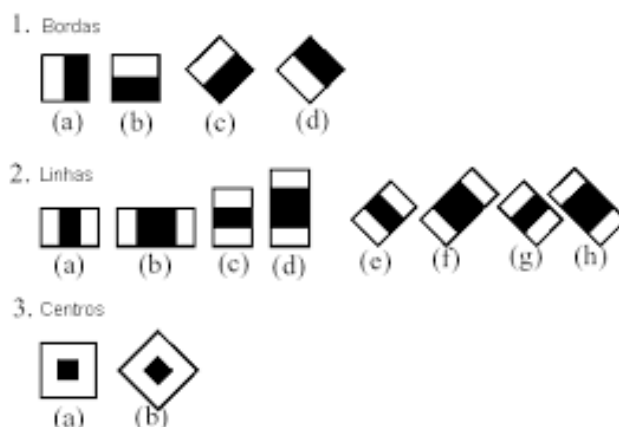
### 3.2 APLICAÇÃO DO ALGORITMO

Esta etapa do desenvolvimento consiste na aplicação do classificador Haar Cascade, para o reconhecimento dos carros dentro de cada vaga, dado um conjunto prévio de treinamento. O trabalho com este tipo de classificador pode ser dividido em duas etapas principais, a primeira diz respeito ao treinamento do algoritmo, que demanda um grande esforço computacional, uma vez que nela são geradas as características que formarão o padrão para detecção do automóvel, sendo esta a segunda etapa do desenvolvimento.

Dentre os algoritmos citados no item 2.1 do referencial teórico, a escolha deste classificador se deu por conta de que em uma comparação simples com o SVM e o HOG, os dois juntos provisionam melhor do que o Haar Cascade sozinho, porém necessitam de muito mais processamento [20], desta forma a escolha deste também reduz os custos na aplicação futura do protótipo em ambiente real.

Para o treinamento do algoritmo é necessário um conjunto de amostras positivas. Neste processo tem de haver imagens que contenham apenas automóveis, o foco principal do trabalho, e um conjunto de imagens negativas, para diferenciar cenários onde não temos nenhum automóvel presente na imagem, até que se encontre a acurácia ideal para satisfazer a aplicação. O conjunto prévio de características consideradas pelo classificador para varrer as amostras em busca das características relevantes, pode-se ver na Figura 5 [20].

Figura 5 - Características do classificador Haar Cascade

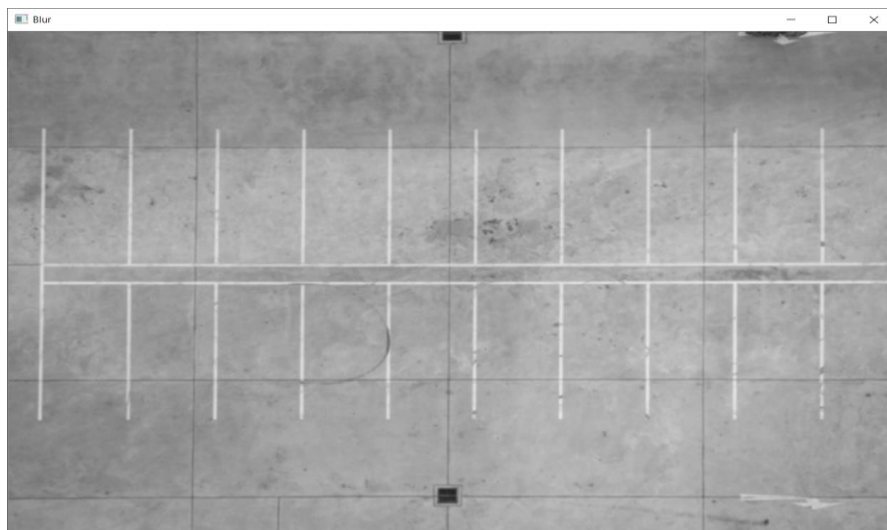


Fonte: baseado em [20].

## 4 ANÁLISE E RESULTADO DOS DADOS

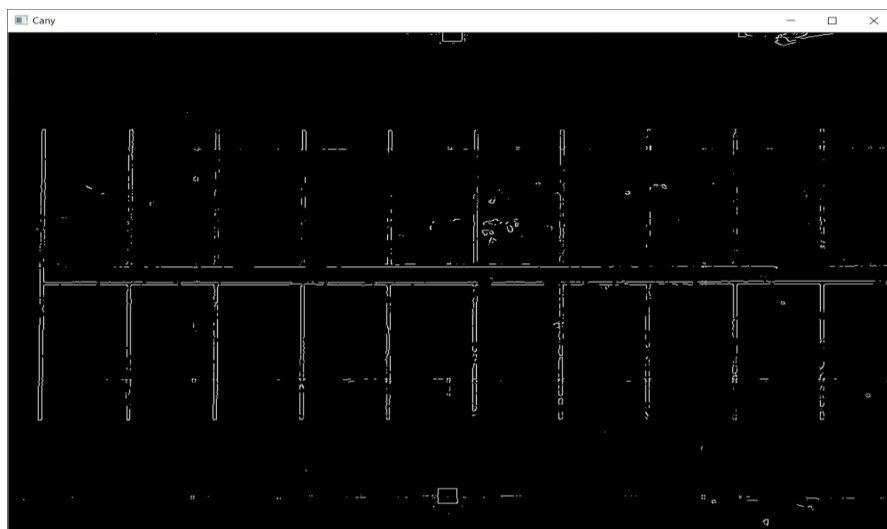
Após coletar as imagens e as submeter aos tratamentos citados no item 3.1, encontram-se os resultados esperados. Na Figura 6, pode-se ver o resultado obtido a partir da limpeza da imagem recebida através da remoção dos ruídos, para a posterior aplicação do Canny, onde se tem o retorno das arestas como demonstrado na Figura 7.

Figura 6 - Resultado da limpeza



Fonte: do Autor.

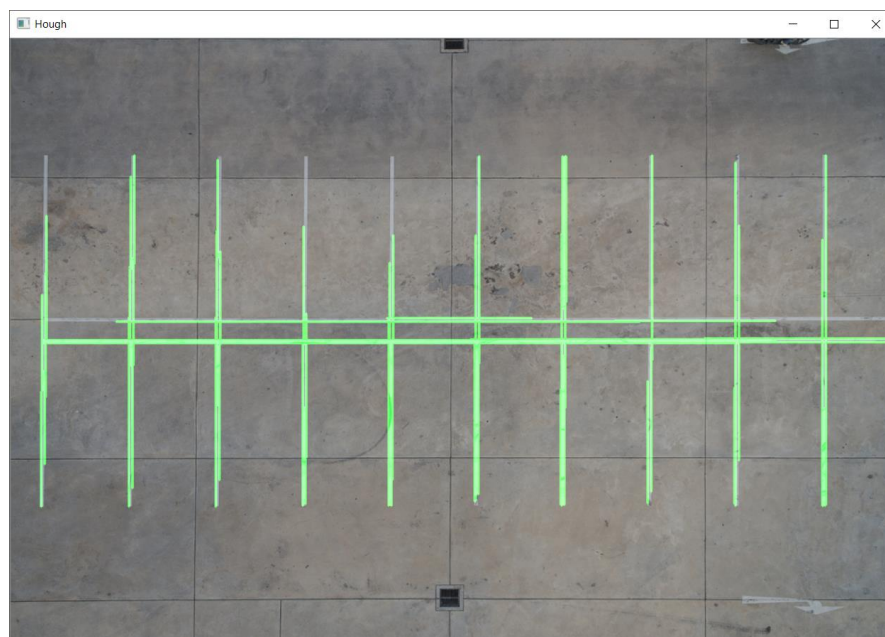
Figura 7 - Resultado da função Canny



Fonte: do Autor.

Após o conhecimento das arestas na imagem se tem o primeiro resultado real, no qual se consegue reconhecer de forma total as vagas do estacionamento recebido por imagem, como demonstrado na Figura 8. Com este reconhecimento se pode delimitar cada vaga, gerando um arquivo com as coordenadas de cada vértice da vaga reconhecida, para posteriormente se identificar o seu status.

Figura 8 - Resultado da transformada de hough

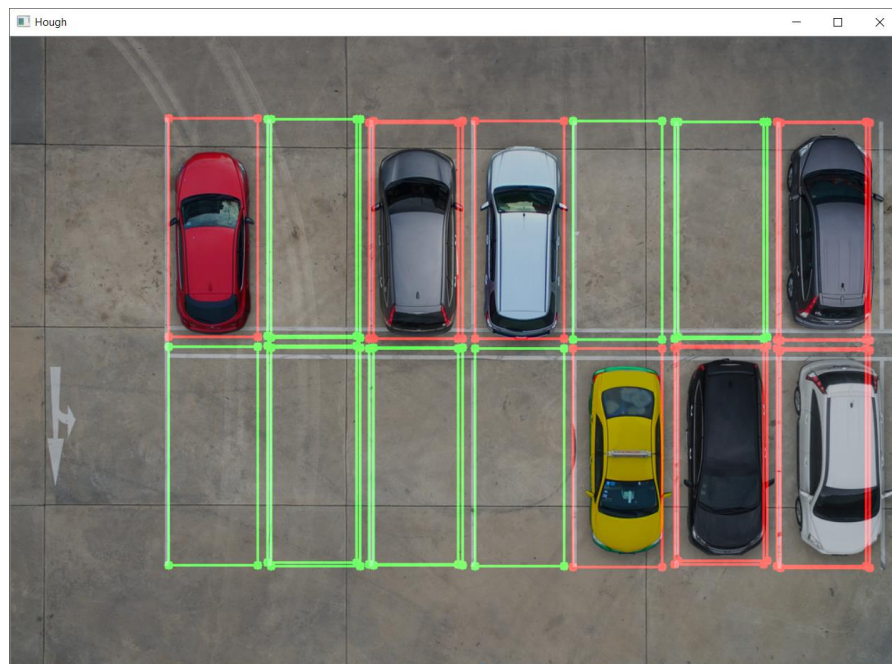


Fonte: do Autor.

Para identificar o status da vaga, pode-se reconhecer os automóveis presentes na imagem e assim identificar se este está presente dentro de alguma vaga previamente reconhecida. Como citado no item 3.2 foi utilizado o algoritmo de classificação Haar Cascade para fazer esta classificação e chegar no resultado esperado. O classificador utilizado e desenvolvido foi treinado em 15 estágios diferentes com 2000 imagens positivas, obtidas a partir de 10 amostras de veículos e com 1900 imagens negativas, utilizando 24x24 pixels, dos quais quanto maior o número, melhor será o resultado do classificador, porém consequentemente imprescindível um maior poder de processamento e mais tempo de treinamento. Com os parâmetros passados para o treinamento se obteve um classificador que respondesse corretamente ao objetivo do projeto.

Quando reconhecidos os automóveis, pode-se gerar um novo arquivo contendo as suas coordenadas (x,y), para assim comparar estas com as coordenadas de cada vaga gerada anteriormente e se chegar ao status da vaga, como mostra a Figura 9, evidenciando as vagas disponíveis em verde e as vagas ocupadas por um veículo com a cor vermelha.

Figura 9 - Resultado do classificador *Haar Cascade*



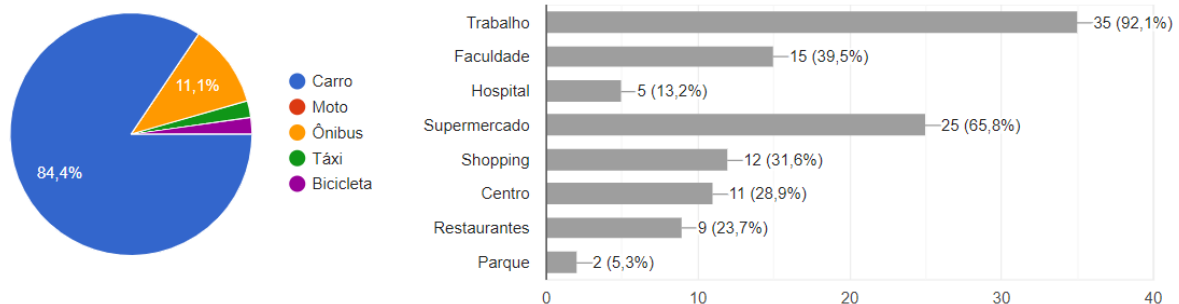
Fonte: do Autor.

Para validação do processo junto aos possíveis usuários finais, foi desenvolvido um formulário através do Google Forms, com perguntas e respostas a respeito do problema que vem sendo abordado neste projeto. O formulário foi espalhado ao público em geral por meio dos grupos de Whatsapp e ficou aberto para respostas durante um dia após a divulgação. Durante este tempo foi possível coletar 45 respostas de pessoas com ideias diferentes, obtendo-se assim informações importantes para validação do problema.

Com relação às questões abordadas no formulário estes focaram em como os usuários costumam se locomover e quais locais da cidade estes costumam frequentar. Assim, abordando os problemas encontrados por eles e como poderia ser solucionado este problema.

Analisando o resultado da primeira etapa do formulário, fica evidente que a maioria das pessoas se locomovem de carro, público este que desejamos atingir. A Figura 10 demonstra este percentual de pessoas que se locomove de carro e quais destinos são frequentemente acessados por elas.

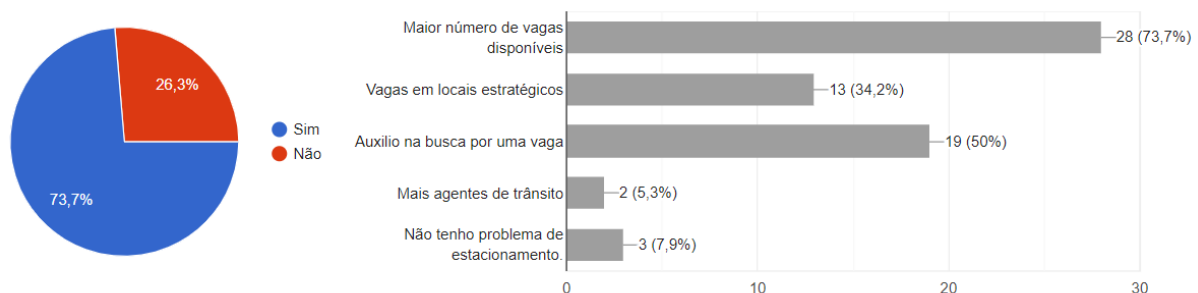
Figura 10 – Forma de locomoção e destinos frequentados



Fonte: do Autor.

Avaliando o destino das pessoas, pode-se perceber que os mais frequentados (trabalho, faculdade e supermercados) são locais que geralmente possuem estacionamento próprio, porém mesmo assim estes encontram dificuldade para estacionar o seu veículo, como demonstrado na Figura 11.

Figura 11 – Dificuldade encontrada para estacionar



Fonte: do Autor.

Para entender o que poderia solucionar a dificuldade encontrada pela maioria das pessoas, ouviu-se a opinião destas e, no topo da pesquisa, como mostra a Figura 11, tem-se o pedido do aumento do número de vagas disponíveis nos locais públicos, seguida da opção que valida o projeto, na qual se colocou à disposição algo



que auxilia na busca por uma vaga. Além destas questões se pede a opinião do público a respeito da solução desta dificuldade, dentre 38 respostas encaminhadas, 13 possuíam relação com o assunto abordado, citando desde o auxílio colocado anteriormente até a própria colocação de um aplicativo, que demonstre as vagas disponíveis em um determinado local.

## **5 CONCLUSÃO**

O objetivo do projeto se atribuiu pelo desenvolvimento de um sistema que auxiliasse os motoristas a encontrar vagas de estacionamento disponíveis, por meio do tratamento de imagens e a aplicação dos conceitos de Inteligência Artificial, visando verificar o estado das vagas no meio das grandes cidades. Para isso foi apresentada uma revisão sobre os conceitos e os modelos mais utilizados no aprendizado de dados, afim de se escolher a melhor forma de implementação do sistema.

A implementação do projeto se deu por meio da linguagem de programação Python, utilizando-se das funções do OpenCV para o processamento das imagens. A finalização aconteceu com o reconhecimento dos automóveis, através do classificador HaarCascade desenvolvido e implementado com a utilização de 2000 amostras positivas treinadas em 15 estágios diferentes, chegando numa taxa de acerto de 99,5% e uma taxa máxima de alarme falso de 50%.

Através da pesquisa de validação do problema, obteve-se o retorno de 45 pessoas e foi possível concluir que a implementação do sistema em um ambiente real, estaria auxiliando e possivelmente melhorando a qualidade de vida das pessoas nos centros das grandes cidades.

Como trabalhos futuros se pode melhorar o próprio classificador, treinando-o com um número maior de amostras de melhor qualidade (mais pixels) e possivelmente mais estágios, e podendo desta maneira utilizar as funções com melhores parâmetros, afim de melhorar o reconhecimento, tanto das vagas quanto dos veículos. Com a pesquisa também se pretende para um trabalho futuro implementar uma aplicação visual com o aproveitamento do algoritmo em um ambiente real, para que se possa validar o sistema com o público alvo do projeto e atingir completamente o objetivo geral proposto no início do projeto.

## 6 AGRADECIMENTOS

Meu primeiro agradecimento vai a Deus, aquele que me permitiu chegar até neste momento vivenciando toda esta experiência.

Agradeço imensamente aos meus pais, João e Maria, por me apoiarem durante toda essa jornada de 6 anos e meio de faculdade, dando sempre muito apoio e me entendendo sempre. Assim como para eles, deixo também o meu muito obrigado à minha namorada Priscila, que desde o início sempre acreditou que daria certo e nunca me deixou desanimar, falando sempre que tudo vai dar certo.

Não poderia deixar de agradecer também aquele que durante dois semestres, sempre esteve disponível para me guiar neste trabalho, meu orientador Anderson Rodrigo Farias, que foi não só um orientador, mas também um incentivador, não me deixando desanimar nas dificuldades encontradas.

Uma pessoa que não poderia deixar de agradecer por ter sido muito importante para a formação da ideia, e que acredito ser quem deu o pontapé inicial para a implementação deste sistema, é o coordenador Gustavo De Lucca, que sempre esteve apoiando o projeto e dando ideias para a melhora deste.

Agradeço também a todos os meus colegas da sala, que juntos chegamos até aqui, enfim, agradeço de coração a todos aqueles que de alguma forma, direta ou indiretamente colaboraram para que este projeto desse certo.

## REFERÊNCIAS

[1] **DETRAN - Departamento Estadual de Trânsito/SC - ESTATÍSTICA**. Disponível em: <<http://www.detran.sc.gov.br/index.php/estatistica/99-estatistica/151-estatistica3333>>. Acesso em: 24 março 2018.

[2] GORGES, L. **Santa Catarina tem a maior taxa de veículos por habitante do Brasil - Diário Catarinense**. Disponível em: <<http://dc.clicrbs.com.br/sc/noticias/noticia/2017/06/santa-catarina-tem-a-maior-taxa-de-veiculos-por-habitante-do-brasil-9808466.html>>. Acesso em: 18 abril. 2018.

[3] PLAMUS. Apresentação Zonas 30 e Ruas Completas – Rede Vida no Trânsito. **Plano de Mobilidade Urbana Sustentável**, 2015.

[4] DE VASCONCELLOS, E. A. Congestionamento no trânsito e financiamento da mobilidade—avaliação dos estudos no Brasil e das perspectivas metodológicas. **Revista dos Transportes Públicos ANTP**, p. 11, 2014.

[5] **ZONA AZUL - CARTÃO AZUL DIGITAL**. Disponível em: <<http://www.mobilidade.com.br/cetsp/>>. Acesso em: 24 março 2018.

[6] SILVA, B. N.; KHAN, M.; HAN, K. Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. **Sustainable Cities and Society**, v. 38, n. Sustain. Cities Soc., p. 697–713, 2018.

[7] BUNDY, A. Preparing for the future of Artificial Intelligence. **Ai & Society**, v. 1, n. 1A, p. 1–48, 2016.

[8] RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. 3<sup>a</sup> ed. New Jersey, United States: Pearson Prentice Hall, 2009.

[9] GUPTA, P. Decision Trees in Machine Learning. **Towards Data Science**, 2017.

[10] ROBERTSON, S. et al. Digital image analysis in breast pathology—from image processing techniques to artificial intelligence. **Translational Research**, v. 194, p. 19–35, 2017.

[11] KULKARNI, A. Multispectral Image Analysis using Decision Trees. **International Journal of Advanced Computer Science and Applications**, v. 8, n. 6, p. 11–18, 2017.

[12] JAAFARI, A.; ZENNER, E. K.; THAI, B. Ecological Informatics Wild fire spatial pattern analysis in the Zagros Mountains , Iran : A comparative study of decision tree based classifiers. v. 43, n. June 2017, p. 200–211, 2018.

[13] SYAM, N.; SHARMA, A. Waiting for a sales renaissance in the fourth industrial revolution: Machine learning and artificial intelligence in sales research and practice. **Industrial Marketing Management**, v. 69, n. November 2017, p. 135–146, 2018.

[14] RAY, S. **Understanding Support Vector Machine algorithm from examples (along with code)**. Disponível em:

<<https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>>. Acesso em: 15 abril 2018.

[15] PARKER, J. R. **Algorithms for Image Processing and Computer Vision**. 2<sup>a</sup> Ed ed. Inidanopolis, Indiana: Wiley Publishing, 2010.

[16] HSU, J. **Finding One Face In a Million - IEEE Spectrum**. Disponível em: <<https://spectrum.ieee.org/computing/software/finding-one-face-in-a-million>>. Acesso em: 15 abril 2018.

[17] **Portfólio de non c no Shutterstock**. Disponível em: <<https://www.shutterstock.com/pt/g/nonnie192>>. Acesso em: 25 abril 2019.

[18] CANNY, J. A Computational Approach to Edge Detection. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. PAMI-8, n. 6, p. 679–698, novembro 1986.

[19] JIANG, S. et al. **Parking Lines Detection Based on an Improved Hough Transform**. (C. Ishii, J. Zhou, G. Capi, Eds.) 2017 International Conference on Robotics and Machine Vision. SPIE, 19 dezembro 2017. Disponível em: <<https://www.spiedigitallibrary.org/conference-proceedings-of-spie/10613/2300492/Parking-lines-detection-based-on-an-improved-Hough-transform/10.1117/12.2300492.full>>. Acesso em: 25 março 2019.

[20] WEI, Y. et al. Multi-vehicle detection algorithm through combining Harr and HOG features. **Mathematics and Computers in Simulation**, v. 155, p. 130–145, janeiro 2019.