

## **Desenvolvimento de sistema para identificação de nível de fadiga e sonolência em condutores de veículos por redes neurais artificiais**

**Giovanni Gonçalves Bovolato**

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – Campus Cubatão, SP, Brasil.

**Guilherme de Almeida Rey Dominguez**

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – Campus Cubatão, SP, Brasil.

**Dr. Arnaldo de Carvalho Junior**

Laboratório Maxwell de Micro-ondas e Eletromagnetismo Aplicado (LABMAX)  
Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – Campus Cubatão, SP, Brasil.

**Resumo:** O número de veículos circulando nas vias cresce a cada ano, trazendo consigo um aumento proporcional nos incidentes de trânsito. Parcela significativa desses acidentes está diretamente ligada à condição de sonolência do condutor. Com o propósito de abordar essa problemática, este estudo se dedica ao desenvolvimento de um sistema baseado em redes neurais artificiais, com o intuito de identificar a fadiga e sonolência em condutores. Em pormenores estão a seleção de uma base de dados para o treinamento da rede e o desenvolvimento de um algoritmo capaz de detectar fadiga e sonolência. Para concretizar esses objetivos, foram utilizadas a linguagem Python e as bibliotecas relacionadas. Um smartphone foi utilizado como câmera e destinava imagens de entrada a um computador que processava todas as rotinas. Através do dataset escolhido uma rede neural foi treinada a fim de classificar o estado do olho entre aberto e fechado. Essa rede somada a outros dois algoritmos, um detector de face e outro detector de olhos, contemplaram o algoritmo final, com a capacidade de classificar o nível de fadiga e classificar o estado de não-atenção.

**Palavras-chave:** Redes Neurais Artificiais. Inteligência Artificial. Detecção de Fadiga. Reconhecimento de Imagem.

**Abstract:** The number of vehicles on the roads increases every year, bringing with it a proportional rise in traffic incidents. A significant portion of these accidents is directly linked to the driver's drowsiness condition. In an effort to address this issue, this study

is dedicated to the development of a device based on artificial neural networks with the purpose of identifying fatigue and drowsiness in drivers. Details include the selection of a database for network training and the development of an algorithm capable of detecting fatigue and drowsiness. To achieve these goals, the Python language and related libraries were employed. A smartphone served as the camera, providing input images to a computer that processed all routines. Through the chosen dataset, a neural network was trained to classify the eye state as open or closed. This network, combined with two other algorithms—a face detector and an eye detector—formed the final algorithm with the ability to classify the level of fatigue and determine the state of inattention.

**Keywords:** *Artificial Neural Networks. Artificial Intelligence. Drowsiness Detection. Image Recognition.*

## INTRODUÇÃO

A cada ano a quantidade de veículos cresce exponencialmente no Brasil, tendo atingido, no ano de 2022, a proporção de um carro a cada três habitantes segundo relatórios do Ministério da Infraestrutura e Censo do Instituto Brasileiro de Geografia e Estatística (IBGE) (Transportes, 2022; IBGE, 2022). O aumento significativo desses contribui assiduamente para maiores quantidades de acidentes automobilísticos.

Segundo o relatório publicado no ano de 2018 pela Organização Mundial da Saúde (OMS), o número de pessoas mortas em decorrência de acidentes de trânsito chegou a cerca de 1,4 milhão (WHO, 2018). Nesse mesmo ano, no Brasil, o número de acidentes chegou em torno de 646 mil, com pouco mais de 990 mil veículos envolvidos (Transportes, 2022) e vitimando mais de 41 mil pessoas (WHO, 2018).

Segundo o Registro Nacional de Acidentes e Estatísticas de Trânsito (RENAEST) o cansaço e sonolência em motoristas de veículos é responsável por grande porcentagem de acidentes de trânsito (Transportes, 2022). De acordo com *Cambridge Dictionary*, fadiga tem como definição “sensação de enfraquecimento” e pode estar relacionada a diversos fatores físicos e psicológicos (Cambridge, 2023).

Estimativas feitas dizem que de 2% a 20% de todas as mortes anuais em acidentes de trânsito estão relacionadas às condições de sono do motorista envolvido.

Segundo a *National Highway Traffic Safety Administration* (NHTSA), em média, por ano, no período de 2009 a 2013, em mais de 72.000 acidentes houve envolvimento de motoristas sonolentos, segundo relatos da polícia, resultando num número de mais de 41.000 feridos e mais de 800 mortos, números obtidos através da FARS (*Fatality Analysis Reporting System*) da NHTSA e da *National Automotive Sampling System* (NASS) (NHTSA, 2016).

Em uma pesquisa feita pela *AAA Foundation for Traffic Safety* analisando os dados do *NASS Crashworthiness Data System* (CDS) da NHTSA, houve a inferência da existência de colisões adicionais com motoristas sonolentos. Através desse estudo, estima-se que 7% de todos os acidentes e 16,5% dos acidentes fatais envolveram um motorista sonolento. A partir dessa estimativa, é possível entender que, somente a partir de acidentes relacionados a sonolência, houve mais de 5.000 mortes nos Estados Unidos no ano de 2015 (NHTSA, 2016; NHTSA, 2017).

Estudos como o da NHTSA também são feitos em outros países, como no Brasil. Em 2019 foi realizada uma pesquisa pela Associação Brasileira de Medicina do Tráfego (ABRAMET) em parceria com o Conselho Regional de Medicina (CRM) e a Academia Brasileira de Neurologia (ABN), a qual indicou que 42% dos acidentes nas ruas e estradas do país são causados pelo sono, enquanto outros 18% são causados por fadiga excessiva. Assim, chega-se à informação de que a direção, quando feita com sono ou cansaço, é responsável por 60% dos acidentes de trânsito no Brasil, número superior aos dados relacionados ao consumo de álcool e drogas (Persono, 2022).

A ABRAMET promoveu também uma campanha, chamada de “Não dê carona ao sono!”, na qual estavam indicados os principais causadores de sonolência ao volante e, conseqüentemente, possíveis acidentes de trânsito, sendo eles a privação de sono; dormir menos de oito horas por dia – sendo o sono entre seis e sete horas por noite um agente que dobra o risco de acidentes e o sono por menos de cinco horas por noite um agente que aumenta o risco de acidentes em 4,5 vezes; ficar acordado por várias horas seguidas; transtornos do sono (Persono, 2022; ANMT, 2017).

Ainda segundo a campanha citada, existem alguns fatores que indicam a viabilidade de um acidente de trânsito por sono ocorrer, sendo eles bocejos ou piscadas frequentes, dificuldade em manter os olhos abertos, visão dupla ou borrada,

dificuldade em manter a cabeça erguida, redução da concentração, dificuldade em manter a velocidade do veículo constante e saída da pista, sendo estas variáveis que, erroneamente, podem ser confundidas com uma possível falta de atenção (ANMT, 2017).

A atenção, crucial para uma direção segura e responsável, pode ser definida como a capacidade de concentrar a mente em um estímulo específico, ignorando outras informações irrelevantes ou distrativas. É um processo cognitivo fundamental que nos permite selecionar e processar informações importantes, filtrando o que é menos importante ou desnecessário em determinado momento. A atenção pode ser voluntária ou involuntária, e é influenciada por fatores como a motivação, a emoção e a fadiga. A atenção é um aspecto essencial da cognição humana, desempenhando um papel importante em muitas atividades cotidianas, como leitura, aprendizado, condução de veículos e interação social.

A detecção precoce de sinais de fadiga e sonolência é fundamental para evitar acidentes e melhorar a segurança nas estradas. Existem diversas formas para identificar a sonolência em seres humanos. Técnicas de monitoramento da atividade cerebral, como o eletroencefalograma (EEG), podem ser utilizadas para identificação. Outra abordagem é a análise da variabilidade da frequência cardíaca, que pode ser realizada de forma não invasiva e apresenta uma correlação positiva com a fadiga. Outrossim, um meio de detecção que tem tido seu uso e estudos fomentados na comunidade científica aumentados exponencialmente é a detecção através de imagens por meio de Redes Neurais Artificiais (RNA) (Philip, Sagaspe, Tailard, 2005; Cajochen, Khalsa, Wyatt, 1999; Hu, Liu, Wang, 2018).

### **Objetivos**

Esse trabalho tem como objetivo principal o desenvolvimento de um sistema baseado em redes neurais para conseguir detectar o estado de “não atenção” de condutores de veículos no menor tempo de resposta possível, utilizando técnicas de RNA.

Como objetivos específicos desse trabalho, estão:

- a) Identificar e selecionar a base de dados a ser utilizada para o treinamento da rede neural;

- b) Desenvolver o algoritmo de RNA e verificar a capacidade de detecção do sistema junto à base de dados utilizada;
- c) Apresentar o correto funcionamento do sistema de Identificação de Nível de Fadiga e Sonolência.

## **REVISÃO DA LITERATURA**

As Redes Neurais Artificiais são uma abordagem científica, através de técnicas computacionais, que busca simular o funcionamento de neurônios, tendo então, como base de desenvolvimento, o cérebro biológico (Instituto, 2023). A partir disso, essas redes baseiam-se em uma conexão entre diversas células que representam os neurônios em um sistema (Bovolato, Dominguez, 2022). O sistema, de forma geral, funciona por meio de diversas ligações que representam entradas e saídas, interligando a saída de uma célula à entrada de outra, para que, dessa forma, as informações sejam processadas por todas as camadas do sistema. As redes neurais artificiais são, dessa forma, um método de processamento de dados feito por todas as células, que são chamadas de neurônios artificiais e, conectados entre si, processam as informações, com grande relação ao funcionamento do cérebro (Bovolato, Dominguez, 2022).

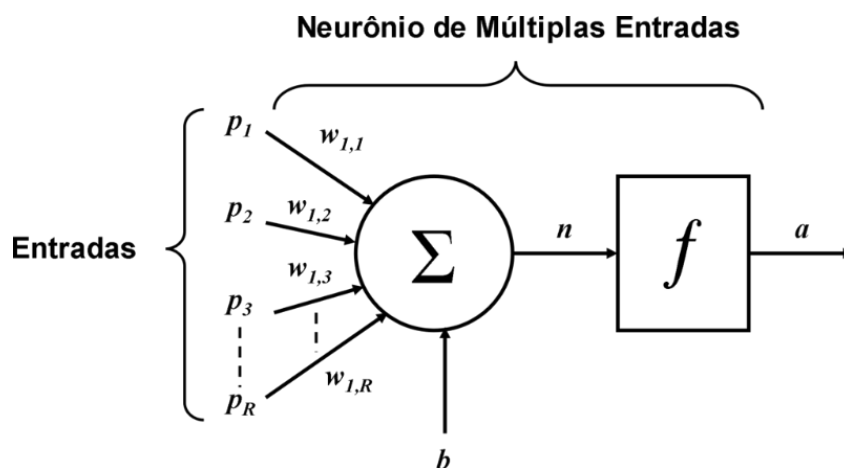
As Redes Neurais possuem capacidades de mapeamento, permitindo que sejam mapeados padrões de entrada para os padrões de saída desejados. Além disso, por ter como possibilidade o aprendizado, essas redes podem ser treinadas a fim de se obter o comportamento desejado por ela. De forma geral, é um sistema muito robusto, sendo resistente a falhas, podendo completar padrões incompletos ou realizar ajustes em padrões que passaram por quebras ou ruídos, por exemplo (Sharma, Chopra, Akshit, 2013).

As RNA, compostas por neurônios artificiais que possuem funções matemáticas inspiradas nos neurônios biológicos, possuem a habilidade de adquirir conhecimento em uma função específica ou na identificação de padrões, despertando interesse no campo de reconhecimento e controle de sistemas (Abiodun, 2018).

Embora não possuam a mesma complexidade de um cérebro, as redes neurais artificiais apresentam duas semelhanças fundamentais com as redes neurais biológicas: a capacidade de descrever seus elementos de construção por meio de

dispositivos computacionais simples e a função da rede é determinada pelas conexões entre os neurônios (Hagan, Demuth, Beale, 2014). Na Figura 1 é representado o diagrama esquemático do modelo matemático do neurônio de uma RNA.

Figura 1 - Diagrama esquemático do modelo matemático do neurônio de uma RNA.



Fonte: Adaptado de (Hagan et al., 2014).

O valor de ajuste, ou *bias*, é representado por  $b$  e é somado a multiplicação da representação da força das sinapses, os pesos, que são identificados por  $w$ . A saída  $a$  é resultado da aplicação de uma função de ativação ( $f$ ) ao resultado da soma ( $n$ ) (De Carvalho Junior, 2021).

O *perceptron* é um algoritmo básico projetado para realizar a classificação em duas categorias distintas. Posto isso, a rede neural artificial que é composta por mais de um *perceptron* denomina-se *multilayer perceptron* (MLP). A sua composição é baseada em uma camada de entrada, responsável por receber o sinal, uma camada de saída, que é capaz de tomar uma decisão acerca da entrada e, entre essas, certa quantidade de camadas ocultas que auxiliam na aproximação de funções contínuas (Academy, 2022). A partir do *perceptron*, diversos neurônios artificiais foram desenvolvidos, com a aplicação de diferentes equações matemáticas, ou funções de ativação ( $f$ ), apresentadas na Figura 1 (De Carvalho Junior, 2021; Carvalho et al, 2022).

A forma em que são dispostas estruturalmente implica em diferentes tipos de arquiteturas para as RNA, existindo as redes neurais recorrentes (RNR), redes

*multilayer perceptrons* (MLP) e as Redes Neurais Convolucionais (RNC) (*convolutional neural network* – CNN), entre diversos tipos de arquiteturas (Academy, 2022).

Após o fornecimento de dados de treino para o algoritmo, as RNA conseguem reconhecer e determinar os padrões existentes em certas situações, que podem variar de uma previsão de compra de clientes através do perfil individual do consumidor até a identificação de sonolência em motoristas de veículos. As RNA possuem vantagens cruciais quando se trata de classificação de imagens, como baixa porcentagem de erro e alta velocidade de resposta, além de haver a possibilidade de um aprimoramento ainda maior nesse tempo, podendo diminuir um processo que é 1/40 de um segundo (0,025s) para 0,008s (Wang, Chen, 2022).

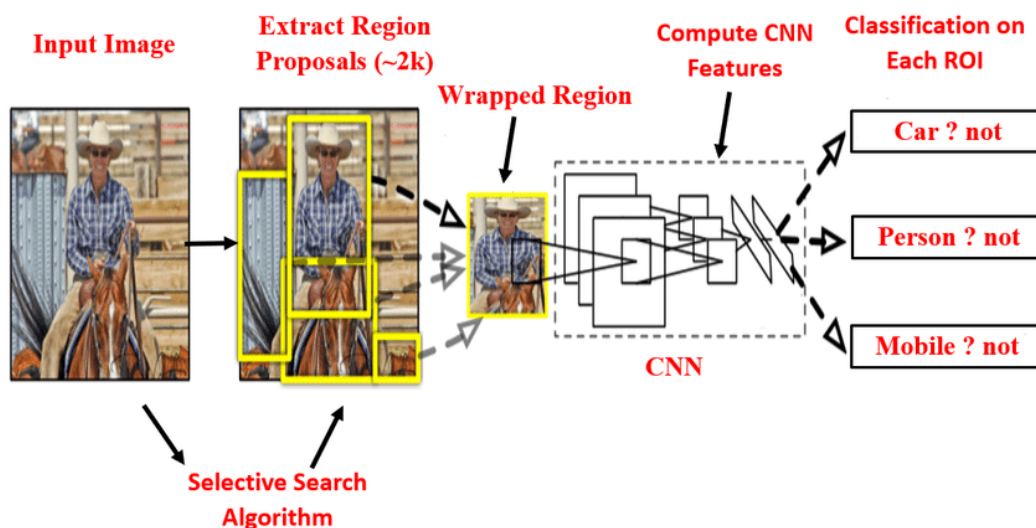
O reconhecimento de imagens conta com adversidades particulares, como deformação, oclusão, desordem de fundo, condições de iluminação, além das variações de ponto de vista, escala e intraclasses (CS231N, 2023). Existem certos métodos de classificação que podem ser utilizados para problemas de classificação, como o *k-Nearest Neighbor Classifier* e a regressão linear, porém sem a confiabilidade e acurácia das redes neurais convolucionais (RNC) (*convolutional neural network* – CNN), ou ConvNets. Uma das principais vantagens das ConvNets é a capacidade de entendimento sobre a complexidade de uma imagem, e, através da utilização de pesos e filtros, consegue captar as dependências temporais e espaciais em uma imagem (Saha, 2018).

O arranjo dos neurônios em três dimensões, altura, largura e profundidade, permite que as RNC tenham uma grande vantagem no reconhecimento de imagens quando comparadas a redes neurais regulares, restringindo a arquitetura de uma forma mais sensível por conta de as entradas serem imagens (CS231N, 2023).

A imagem da Figura 2 a seguir demonstra, de maneira lúdica, o funcionamento de uma RNC, desde a entrada da imagem, a extração de propostas de região, a região envolvida e a área de atuação da RNC, para que então se obtenha a classificação para a imagem proposta.



Figura 2 - Arquitetura de uma RNC.



Fonte: Adaptado de (Murthy, 2020).

Python, como uma linguagem de alto nível, tem desempenho inferior em tarefas de computação intensiva quando comparada a linguagens de nível inferior. Porém, para compensar essa falta de desempenho, existem diversas bibliotecas complementares, feitas a partir de linguagens inferiores, como C, que contemplam conteúdos que auxiliam o desenvolvimento de algumas áreas através do Python. A ampla utilização dessa linguagem permite que sejam, a cada dia, desenvolvidas novas bibliotecas e códigos colaborativos, impulsionando ainda mais sua aplicação e compensando seus pontos negativos (Raschka, 2016).

## METODOLOGIA

Neste capítulo serão apresentados a metodologia, materiais e algoritmos utilizados para a realização deste trabalho.

### Componentes

O elemento principal desse projeto é o computador no qual será processado o algoritmo desenvolvido. Através dele será feito o reconhecimento das imagens captadas, que por sua vez serão processadas e classificadas, gerando a detecção e sinais pretendidos.

O computador utilizado possui as seguintes configurações:



Tabela 1 - Configurações do Computador.

COMPUTADOR	
Processador	Ryzen 5 5600x
Placa Mãe	Asus B450M
Placa de Vídeo	Nvidia GTX 1050 Ti
Memória RAM	16GB
Armazenamento	SSD 480GB

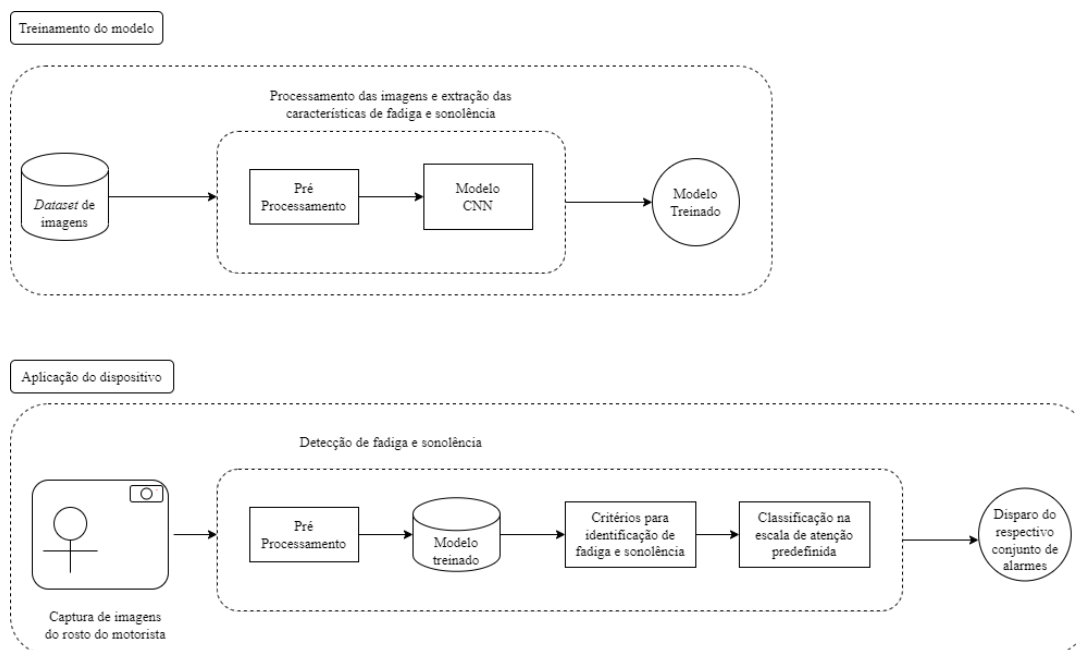
Fonte: Dos Autores (2023).

Neste trabalho, um *smartphone* foi utilizado como uma câmera externa ao computador que, ao captar imagens em tempo real, as transmite ao computador para que o algoritmo seja processado utilizando as imagens produzidas pela câmera do *smartphone*.

### Diagrama de Blocos

Com base nos componentes adquiridos e a arquitetura planejada inicialmente para o desenvolvimento desse trabalho foi montado o seguinte diagrama de blocos:

Figura 3 - Diagrama de Blocos do projeto.



Fonte: Dos Autores (2023).

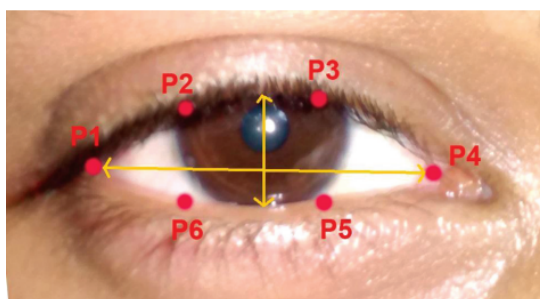
### Trajetos da Imagem

A captura de imagens é realizada em tempo real através da câmera traseira do *smartphone*. A conexão dela com o computador é feita através de um endereço IP da câmera, que possibilita uma conexão remota a ela.

A classificação de fadiga de um condutor é feita através do conceito de *blink rate*. Essa concepção, mais conhecida como taxa de piscadas, é baseada na quantidade de piscadas realizada pelo ser humano no período de um minuto. Essa taxa, em um ser humano em estado habitual de atenção, varia de 8 a 10 piscadas por minuto. O *blink rate* de um indivíduo em estado de sonolência varia de 4 a 6 piscadas (Yusri, Mangat, Wasenmüller, 2021).

A classificação de fadiga através do conceito do *blink rate* só é possível por conta do *Eye Aspect Ratio* (EAR). O EAR é a detecção proporcional do olho, que capta as diferenças de dimensões existentes entre olhos abertos e fechados. Essas diferenças aparecem principalmente na relação entre altura e largura do olho, e essas são identificadas através de pontos predefinidos de 1 a 6, mostrados na Figura 4 (Islam, Rahaman, Ahad, 2019).

Figura 4 - 6 Coordenadas de um olho representadas como P1 a P6.



Fonte: Adaptado de (Islam, Rahaman, Ahad, 2019).

De acordo com o trabalho de Soukupova & Cech 2016, o EAR pode ser representado matematicamente através da Eq. 1:

$$EAR = \frac{\|P2 - P6\| + \|P3 - P5\|}{2\|P1 - P4\|} \quad (1)$$

De acordo com a Eq. 1, o valor do EAR chega próximo a zero quando uma piscada ocorre. Segundo estudos realizados pelo *Harvard Database of Useful*

*Biological Numbers*, uma piscada tem duração entre 100 e 400 milissegundos (B10NUMB3R5, 2023). Com a definição do tempo de uma piscada se torna possível diferenciar uma piscada natural do fechamento dos olhos. Como a piscada tem duração máxima de 0,4 segundos, quando esse tempo excede o período de 0,5 segundos, esse ato é considerado como um fechamento de olhos, determinando, dessa forma, que o motorista se encontra fadigado e sonolento (Svensson, 2004).

### **Base de dados**

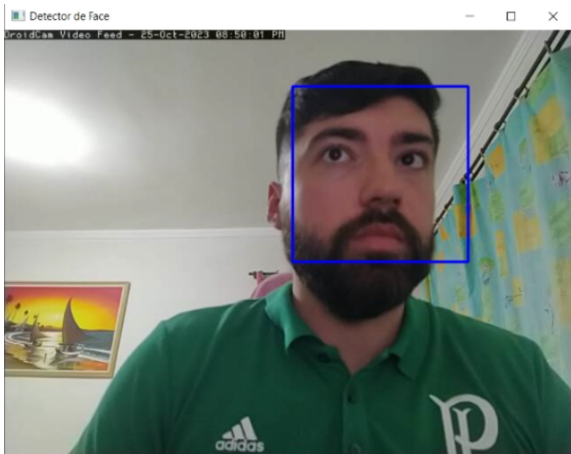
Para a base de dados do projeto, dois *datasets* do site Kaggle foram selecionados, um monocromático em tons de cinza e o outro em cores, sendo um do usuário Dheeraj Perumandla (Kaggle, 2023) e o outro de Prasad V Patil (Kaggle, 2023).

### **Algoritmo 1 – Detector de Face**

A primeira sub-rotina utilizada no trabalho tem como objetivo a detecção da face de uma pessoa. Inicialmente são importadas as bibliotecas Open CV e os. O arquivo XML então é importado através do uso da biblioteca os, que permite a interação do algoritmo com o sistema operacional. Com o arquivo importado, é iniciada a classificação através do uso da *webcam* ou câmera externa utilizada. A leitura do conteúdo é feita em *loop*, sempre buscando a captura do próximo *frame*.

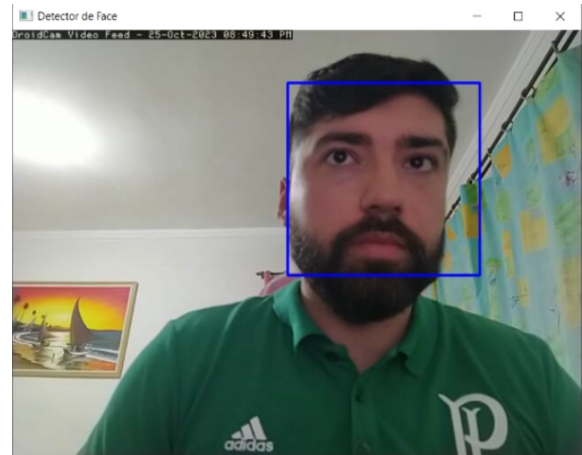
As imagens obtidas através do dispositivo de gravação são convertidas para tons de cinza e, então, é feita a detecção de onde está a face da pessoa no *frame* capturado. Com a classificação é desenhado um retângulo em volta da face na imagem plotada, de forma a ser possível visualizar a classificação realizada no *frame* analisado. A seguir estão exemplos de detecção da face obtidos através deste algoritmo:

Figura 5 – Exemplo 1 de detecção de face.



Fonte: Dos Autores (2023).

Figura 6 – Exemplo 2 de detecção de face



Fonte: Dos Autores (2023).

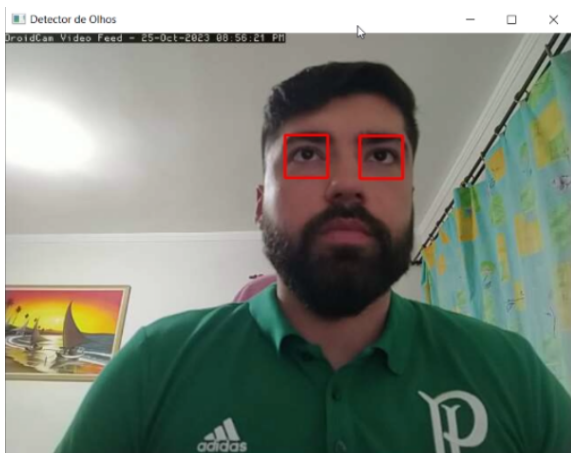
### **Algoritmo 2 – Detector de Olhos**

A segunda sub-rotina do algoritmo principal do trabalho tem como objetivo a localização dos olhos de uma pessoa. Esse método, extremamente semelhante ao primeiro algoritmo, funciona a partir da importação de um arquivo do tipo XML como base de dados. As mesmas bibliotecas são importadas e utilizadas ao decorrer de todo o algoritmo, e a classificação feita através de uma câmera, com a leitura do conteúdo sendo feita em *loop*.

Com a classificação são desenhados retângulos em volta dos olhos na imagem plotada, de forma a ser possível visualizar a classificação realizada no *frame* analisado. Esse algoritmo tem a capacidade de realizar a detecção dos olhos na face quando a pessoa está utilizando óculos, ampliando sua utilização.

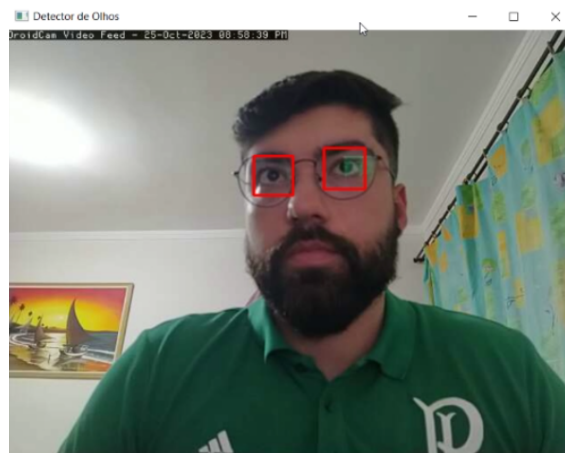
A seguir estão exemplos de detecção da face obtidos através deste algoritmo:

Figura 7 - Detecção dos olhos (sem óculos).



Fonte: Dos Autores (2023).

Figura 8 - Detecção dos olhos (com óculos).



Fonte: Dos Autores (2023).

### Algoritmo 3 – Detector de Estado do Olho

A terceira sub-rotina utilizada no projeto tem como objetivo a classificação do estado do olho entre “aberto” e “fechado”. Diferentemente dos dois primeiros algoritmos apresentados, essa rede neural foi treinada utilizando um banco de imagens, apresentado anteriormente como base de dados.

Foi utilizado um *dataset* com um total de 4 mil imagens, sendo 80% destinado ao treinamento e 20% à validação. As imagens passaram por pré-processamentos antes de serem utilizadas no treinamento e na validação. Foi aplicado redimensionamento em todas elas, para um tamanho de 24 por 24 *pixels*, além de serem convertidas em tons de cinza.

O treinamento aplicado usou 30 épocas de treinamento e atingiu uma acurácia de 99,8% na rede. A seguir está um exemplo de código de teste utilizado com a rede treinada e o resultado obtido, classificando corretamente o estado do olho em uma imagem.

Figura 9 - Algoritmo de teste utilizando a rede treinada para classificar o estado do olho.

```
import random as r
rnd=r.randrange(99)
imagemOriginal = cv2.imread(diretorio+'/teste/_723.jpg')

imagem = cv2.resize(imagemOriginal, (32,32))
imagem = np.expand_dims(imagem, axis=0)

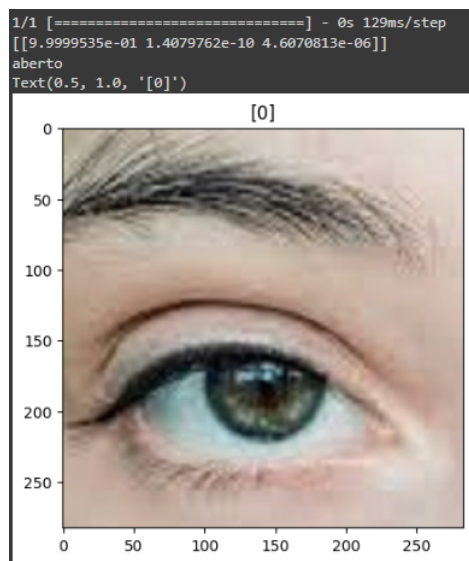
predicao = rede.predict(imagem)
print(predicao)
classificacao = np.argmax(predicao, axis=-1)

if classificacao == 0:
    print('aberto')
else:
    print('fechado')

plt.imshow(cv2.cvtColor(imagemOriginal, cv2.COLOR_BGR2RGB))
plt.title(classificacao)
```

Fonte: Dos Autores (2023).

Figura 10 - Resultado do teste utilizando a rede neural treinada.



Fonte: Dos Autores (2023).

#### Algoritmo 4 – Detector de Fadiga

A construção da rede final utilizada é embasada nas três redes anteriores trabalhando em conjunto para atingir o seu objetivo final, que é a identificação do nível de fadiga do condutor através do estado dos seus olhos.

O algoritmo, portanto, está sempre buscando encontrar um rosto humano, após isso, localizar os seus olhos e, assim, identificando o estado em que ele se encontra para caracterizar ou não fadiga e sonolência no motorista que está à frente da câmera.

Através das seguintes linhas de código, na imagem abaixo, pode-se observar a importação dos classificadores via XML.

Figura 11 - Trecho de código que realiza a importação dos XML como classificadores de suas respectivas funções.

```
rosto =  
cv2.CascadeClassifier(f'{os.getcwd()}\\Data\\drowsiness_data\\haarc  
ascade_frontalface_alt.xml')  
  
olhoesq =  
cv2.CascadeClassifier(f'{os.getcwd()}\\Data\\drowsiness_data\\haarc  
ascade_lefteye_2splits.xml')  
  
olhodir =  
cv2.CascadeClassifier(f'{os.getcwd()}\\Data\\drowsiness_data\\haarc  
ascade_righteye_2splits.xml')
```

Fonte: Dos Autores (2023).

Seguindo a linha de reprodução do algoritmo, pode-se chegar até o ponto em que, dentro de um laço de repetição for, é realizada a predição do estado do olho, tanto o esquerdo, quanto o direito e retornam o resultado para suas respectivas variáveis.

Figura 12 - Trecho do código que realiza a predição dos olhos direito e esquerdo.

```
dirpred = np.argmax(rede.predict(olho d), axis=1)  
esqpred = np.argmax(rede.predict(olho e), axis=1)
```

Fonte: Dos Autores (2023).

Depois de possuir o valor do estado de cada olho armazenado em suas variáveis, a rede então parte para uma validação através de um *if*, que analisa se ambos os olhos se encontram fechados e, caso isso seja verdade, soma pontos para o score atual de Nível de Fadiga.

Figura 13 - Trecho de código que realiza a validação de ambos os olhos e aumento do nível de fadiga.

```
if (dirpred[0] == 0 and esqpred[0] == 0):  
    valor += 1  
    cv2.putText(frame, "FECHADO", (10, height - 20), fonte, 1,  
(255, 255, 255), 1, cv2.LINE_AA)
```

Fonte: Dos Autores (2023).



Passando por toda a classificação, é realizada então a verificação se o valor atual do Nível de Fadiga se encontra abaixo ou acima do limite predefinido pelos autores e a partir dessa análise são disparados os alarmes sonoros e luminosos caso sejam necessários.

Figura 14 - Trecho de código que realiza análise do Nível de Fadiga atual x o Nível de Fadiga tolerado.

```
if (valor > limite):  
    ts = time.time()  
  
    st = datetime.datetime.fromtimestamp(ts).strftime('%d-%m-%Y_%H-%M-%S')  
  
    nome_arquivo = str("Olho Fechado") + " " + str(st)  
  
    cv2.imwrite(os.path.join(path, + str(file_name) +  
'frame.jpg'), frame)  
  
    try:  
        alarme.play()  
    except:  
        pass  
  
    if (espessura < 16):  
        espessura += 2  
    else:  
        espessura -= 2  
        if (espessura < 2):  
            espessura = 2  
  
    cv2.rectangle(frame, (0, 0), (width, height), (0, 0, 255),  
thickness=espessura)
```

Fonte: Dos Autores (2023).

Além do gatilho de alarmes, o algoritmo também permite realizar a captura do *frame* quando o condutor demonstrou fadiga acima do nível tolerado e salva essa captura de imagem junto da data e horário no qual ela foi realizada.

## RESULTADOS

Como dito anteriormente, o algoritmo detector de fadiga é uma junção das outras três redes em funcionamento simultâneo. Dessa forma, neste capítulo serão apresentados os resultados obtidos através de testes realizados com esse algoritmo,

utilizando a câmera do *smartphone* como imagem de entrada e o computador como local de processamento.

A imagem a seguir mostra o algoritmo em funcionamento, com um círculo delimitando a face, onde será detectado o local dos olhos e, posteriormente, o estado dos olhos, se “aberto” ou fechado”. Nela é possível observar que o algoritmo detecta corretamente, mesmo com o uso de óculos, o estado dos olhos – aberto – e mede o nível de fadiga como 0.

Figura 15 - Algoritmo detectando olho aberto e nível de fadiga 0.



Fonte: Dos Autores (2023).

Figura 16 - Algoritmo detectando piscada (olho fechado durante a piscada, nível de fadiga sobe para 1).

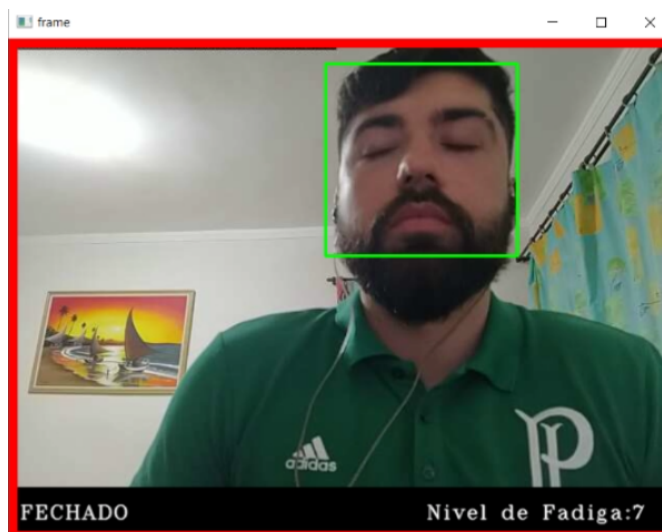


Fonte: Dos Autores (2023).

Ao piscar, o algoritmo faz a detecção da alteração do estado do olho de aberto para fechado, e soma um nível à classificação de fadiga, passando de 0 para 1.

A cada *frame* consecutivo analisado pela rotina e identificado como olho fechado é somado uma unidade ao nível de fadiga. Caso esse nível chegue a 7, é considerada não mais uma piscada, mas sim um fechamento dos olhos. Com essa classificação é acionado um alarme visual (retângulo vermelho intermitente) e um alarme sonoro.

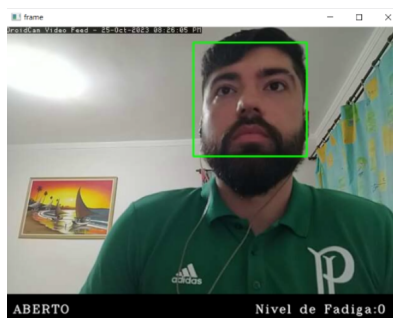
Figura 17 - Algoritmo detectando um fechamento de olhos - estado de não-atenção - e disparando alarmes.



Fonte: Dos Autores (2023).

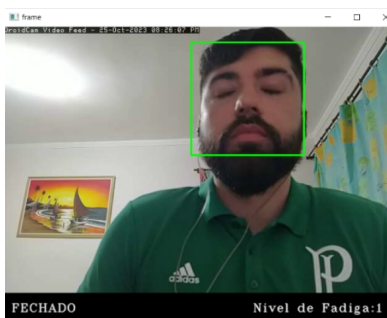
A seguir serão mostrados três *frames* consecutivos, sendo o primeiro deles com o olho aberto, o segundo realizando uma piscada e o terceiro mantendo o olho fechado após a piscada.

Figura 18 - Olho aberto, estado de atenção.



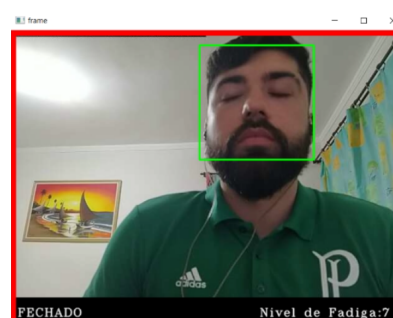
Fonte: Dos Autores (2023).

Figura 19 - Olho fechado, piscada comum.



Fonte: Dos Autores (2023).

Figura 20 - Estado de não-atenção, subindo o nível de fadiga para 7 e ativando alarme.



Fonte: Dos Autores (2023).

Devido às limitações do formato do trabalho escrito, somente imagens puderam ser incluídas. No entanto, para proporcionar um exemplo visual e comprovação de resultados, um vídeo demonstrativo está disponível em (Algoritmo, 2023).

## CONCLUSÃO

Através dos estudos realizados foi validado o conceito de detecção de fadiga através de RNA. A utilização da linguagem de programação Python e suas bibliotecas possibilitou a utilização de algoritmos que foram desde o treinamento de redes neurais até sua utilização.

Os algoritmos, individualmente, colaboraram para a rotina final que possui a capacidade de detectar a face, a localização dos olhos e classificar o estado dos olhos entre aberto e fechado. Através disso, a análise do estado de atenção foi possibilitada e aplicada de forma efetiva e com bons resultados em todos os testes feitos e apresentados.

Dessa forma, entende-se que os objetivos propostos inicialmente foram atingidos com excelência, visto que a base de dados utilizada no treinamento da rede neural foi suficiente para se obter uma alta taxa de acurácia, além do pleno funcionamento do sistema, classificando corretamente os níveis de fadiga e rotulando o estado de não-atenção quando detectado.

## REFERÊNCIAS

ABIODUN, O. I., et al. State-of-the-art in artificial neural network applications: A survey. **Heliyon**, Volume 4, Issue 11, p. 41, 2018.

ACADEMY, D.S. Deep Learning Book. **Data Science Academy**, 2022. Disponível em: <https://www.deeplearningbook.com.br/>. Acesso em: 25 Mai. 2023.

ALGORITMO | Detector de Fadiga. Direção: Giovanni Bovolato. Intérprete: Guilherme Dominguez. **YouTube**, São Paulo, 2023. Disponível em: [https://www.youtube.com/watch?v=\\_e9Efj4Jx88&ab\\_channel=GiovanniBovolato](https://www.youtube.com/watch?v=_e9Efj4Jx88&ab_channel=GiovanniBovolato). Acesso em: 15 nov. 2023.

ANMT. Não dê carona ao sono!: a fadiga e o sono como causa de acidentes. **ANAMT**, 17 mar. 2017. Disponível em: <https://www.anamt.org.br/portal/2017/03/17/nao-de-carona-ao-sono-a-fadiga-e-o-sono-como-causa-de-acidentes/>. Acesso em: 09 Abr. 2023.

AUTOPAPO. Mais de 40% dos acidentes de trânsito acontecem por sonolência, afirma a Abramet. **Uol**, 06 set. 2019. Disponível em: <https://autopapo.uol.com.br/noticia/mais-de-40-dos-acidentes-de-transito-acontece-por-sonolencia-afirma-a-abramet/>. Acesso em: 08 abril 2023.

B10NUMB3R5. Average duration of a single eye blink. **The Database of Useful Biological Numbers - Harvard University**. Disponível em: <https://bionumbers.hms.harvard.edu/bionumber.aspx?id=100706&ver=0>. Acesso em 10 Ago. 2023.

BOVOLATO, Giovanni; DOMINGUEZ, Guilherme de Almeida. Classificação de Resíduos Sólidos Por Redes Neurais Artificiais Utilizando Edge Impulse. **IFSP**. Cubatão-SP. 2022.

CAJOCHEN C.; KHALSA S.B.; WYATT J.K., et al. EEG and ocular correlates of circadian melatonin phase and human performance decrements during sleep loss. **Am J Physiol**. 1999;277(3 Pt):R640-R649.

CAMBRIDGE. Fatigue. In: **Cambridge Dictionary**, 2023. Disponível em: <https://dictionary.cambridge.org/dictionary/english/fatigue>. Acesso em: 02 abr. 2023.

CS231N. Convolutional Neural Networks. **CS231N - Git Hub**. Disponível em: <https://cs231n.github.io/convolutional-networks/>. Acesso em: 15 jun. 2023.

CARVALHO, A., JUSTO, J.F., ANGELICO, B.A. *et al.* Model reference control by recurrent neural network built with paraconsistent neurons for trajectory tracking of a rotary inverted pendulum, **Applied Soft Computing**, 2022, 109927, ISSN 1568-4946. DOI: 10.1016/j.asoc.2022.109927

CS231N. Image Classification. **CS231n: Deep Learning for Computer Vision**, Stanford – Spring, 2023. Disponível em: <https://cs231n.github.io/classification/>. Acesso em: 5 abr. 2023.

DE CARVALHO JUNIOR, A. Identificação e Controle de Sistemas Dinâmicos com Rede Neural Paraconsistente. 196 p. Tese (Doutorado) – Programa de Engenharia Elétrica, **Escola Politécnica, Universidade de São Paulo**, São Paulo, 2021. DOI: 10.11606/T.3.2021.tde-08102021-100149.

HAGAN, M. T.; DEMUTH, H. B.; BEALE, M. H. Neural Network Design. **Martin Hagan**; 2<sup>o</sup> edition, 2014, p. 802.

HU, S.; LIU, H.; WANG, D., et al. Noninvasive evaluation of mental fatigue using visible light-induced near-infrared spectroscopy. **Sci Rep**. 2018;8(1):16941.

IBRE. Censo Demográfico. **Instituto Brasileiro de Geografia e Estatística**, 2022. Disponível em: <https://www.ibge.gov.br/estatisticas/sociais/rendimento-despesa-e-consumo/22827-censo-demografico-2022.html>. Acesso em: 31 mar. 2023.

INSTITUTO, Ciências Matemáticas. Redes Neurais Artificiais. **USP - Instituto De Ciências Matemáticas E De Computação**. Disponível em: <https://sites.icmc.usp.br/andre/research/neural/>. Acesso em: 10 abr. 2023.

ISLAM, Arafat; RAHAMAN, Naimur; AHAD, Md Atiqur Rahman. A study on tiredness assessment by using eye blink detection. **Jurnal Kejuruteraan**, v. 31, n. 2, p. 209-214, 2019.

KAGGLE. Drowsiness dataset. **Kaggle**, 2023. Disponível em: <https://www.kaggle.com/datasets/dheerajperumandla/drowsiness-dataset> Acesso em 25 ago. 2023.

KAGGLE. Drowsiness Detection Dataset. **Kaggle**, 2023. Disponível em: <https://www.kaggle.com/datasets/prasadvpatil/mrl-dataset> Acesso em 27 ago. 2023.

MURTHY, Chinthakindi Balaram et al. Investigations of object detection in images/videos using various deep learning techniques and embedded platforms—A comprehensive review. **Applied sciences**, v. 10, n. 9, p. 3280, 2020.

NHTSA Drowsy Driving Research and Program Plan. **National Highway Traffic Safety Administration**, mar. 2016.

NHTSA. Asleep at the Wheel – A national Compendium of Efforts to Eliminate Drowsy Driving. **National Highway Traffic Safety Administration**, mar. 2017.

NUMPY. **Site do Numpy**, c2023. Home. Disponível em: <<https://numpy.org/>>. Acesso em: 23 jun. 2023.

PERSONO. Dirigir com sono causa mais acidentes que o álcool. **Persono**, 05 abr. 2022. Disponível em: <https://persono.com.br/insights/saude-e-bem-estar/maio-amarelo-dirigir-com-sono>. Acesso em: 10 abr. 2023.

PHILIP, P.; SAGASPE, P.; TAILARD, J., et al. Fatigue, sleepiness, and performance in simulated versus real driving conditions. **Sleep**. 2005;28(12):1511-1516.

RASCHKA, Sebastian. Python Machine Learning. **Pennsylvania: Packt Publishing**, 2016.

SAHA, Sumit. A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way. **Towards Data Science**, 2018. Disponível em: <<https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>>. Acesso em: 5 de abr. de 2023.

SHARMA, Ayushi; CHOPRA, AKSHIT. Artificial neural networks: Applications in management. **Journal of Business and Management**, v. 12, n. 5, p. 32-40, 2013.

SVENSSON, U. Blink behaviour based drowsiness detection – method development and validation. **Master's thesis project in Applied Physics and Electrical Engineering, Reprint from Dept. Biomedical Engineering, Linköping University, 2004**.

TENSORFLOW. **Site do TensorFlow**. Home. Disponível em: <<https://www.tensorflow.org/?hl=pt-br>>. Acesso em: 20 jun. 2023.

TRANSPORTES, Ministério. Frota de Veículos - 2022. **Ministério da Infraestrutura**, 2 mar. 2022. Disponível em: <https://www.gov.br/infraestrutura/pt-br/assuntos/transito/conteudo-Senatran/frota-de-veiculos-2022>. Acesso em: 28 mar. 2023.

TRANSPORTES, Ministério. Registro Nacional de Acidentes e Estatísticas de Trânsito. **Ministério da Infraestrutura**, 26 set. 2021. Disponível em: <https://www.gov.br/transportes/pt-br/assuntos/transito/arquivos-senatran/docs/renaest>. Acesso em: 31 mar. 2023.

WANG, Peng; CHEN, Meng-ya. Fast recognition of cat and dog images based on network compression. In: **Second International Symposium on Computer Technology and Information Science (ISCTIS 2022)**. SPIE. p. 694-699, 2022.

WHO. Global status report on road safety 2018. World Health Organization, 17 jun. 2018. Disponível em: <https://www.who.int/publications/i/item/9789241565684>. Acesso em: 31 mar. 2023.

YUSRI, Muhammad Fawwaz; MANGAT, Patrick; WASENMÜLLER, Oliver. Detection of Driver Drowsiness by Calculating the Speed of Eye Blinking. **arXiv preprint arXiv:2110.11223**, 2021.