

Termômetro Infravermelho e detector de máscaras faciais com Raspberry Pi

Ludmylla Martins Caetano

UnB - FGA

Tucumã-PA, Brasil

Email: ludmyllacaetano01@gmail.com

Marcus Vinícius Teodoro Mendonça

UnB - FGA

Gama-DF, Brasil

Email: marcus.viniciust07@gmail.com

Resumo—Este artigo relata os processos realizados na construção de um termômetro infravermelho e detector de máscaras faciais utilizando uma Raspberry Pi. O produto, a ser utilizado em estabelecimentos comerciais de pequeno porte, cadastra os funcionários e realiza a medição periódica da temperatura através de um sensor infravermelho e a correta utilização da máscara através de reconhecimento facial. Os dados são armazenados e fornecem um relatório periódico dos funcionários.

I. INTRODUÇÃO

O Covid-19, SARS-CoV-2, é uma síndrome respiratória aguda grave. Sua primeira aparição ocorreu em 2019, na China, tendo se alastrado para o restante do mundo. A origem da virose ainda é incerta, mas é creditada ao morcego, devido a sua semelhança com outras doenças causadas pelo animal.

Seus sintomas incluem tosse, espirro e febre, sendo semelhantes aos de uma gripe, podendo, porém, evoluir para um caso mais sério de pneumonia. Além disso, o vírus afeta de maneira mais severa pessoas afetadas por outras doenças ou com um sistema imunológico mais fragilizado.

Com a pandemia diversas medidas de contenção foram estabelecidas para evitar a propagação do vírus. Dentre as medidas estão o uso de máscaras e a checagem da temperatura corporal.

O funcionário de um estabelecimento comercial que se mantém em contato com pessoas, clientes ou colegas, durante grande parte do expediente deve estar sempre utilizando uma máscara facial. Para preservar a saúde desse funcionário a medição periódica de sua temperatura é uma maneira de verificar se não há nenhuma alteração na saúde do mesmo. Assim para a automatização desse processo, utilizaremos uma Raspberry Pi em conjunto com os devidos sensores mais viáveis para a aplicação.

A Raspberry Pi é um microcomputador, que apesar de seu tamanho diminuto, possibilita diversas aplicações. Assim, o seu uso combinado a um sensor infravermelho, para detectar a temperatura, e uma câmera, para detectar o uso de máscara, se torna uma ferramenta extremamente viável na checagem das corretas utilizações das medidas de contenção ao Covid-19.

A. Justificativa

Uma das principais formas de disseminação do vírus é através de espirros e tosses. Em ambientes fechados e de tamanho reduzido é difícil manter o distanciamento social,

assim o uso de máscara facial se torna essencial na redução da propagação da doença, por funcionar como uma barreira contra o vírus. Porém, seu uso, muitas vezes é esquecido ou negligenciado, por uma parcela da população, tornando necessária uma verificação do correto uso das mesmas no início e no final do expediente de trabalho.

Um dos principais sintomas da Covid-19 é a febre, assim a detecção da temperatura é um excelente indício para uma verificação inicial da doença. Não assegura totalmente se o indivíduo possui o vírus, mas já é uma maneira de se tomar as primeiras medidas.

Além disso, o monitoramento periódico de dados relacionados a saúde dos funcionários auxilia em um início mais rápido no tratamento da doença.

B. Objetivos

O projeto busca assegurar uma maior segurança para os funcionários do estabelecimento onde o produto será instalado. O processo será todo automatizado e os dados serão enviados para o responsável pela saúde dos funcionários da empresa, além de que o funcionário também terá acesso aos seus respectivos dados.

C. Escopo

O projeto será voltado para empresas de tamanho médio, com funcionários em contato direto com clientes ou muitos colegas de trabalho. Cumprirá os requisitos:

- Verificação da utilização de máscara;
- Medição da temperatura;
- Armazenamento de dados;
- Envio de dados/relatório;

As checagens serão realizadas, preferencialmente, no início e no final do expediente de trabalho.

D. Requisitos

Os requisitos necessários para o projeto são:

- Raspberry Pi;
- Sensor de temperatura modelo MLX90614DAA com precisão de ± 0.1 °C;
- Módulo de Câmera ver 1.3;

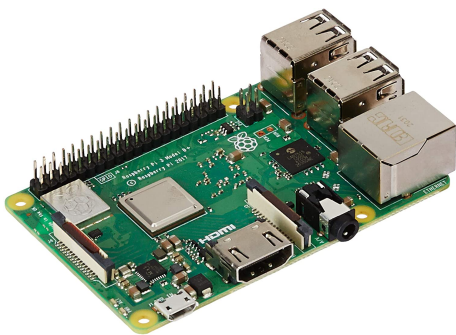


Figura 1. Raspberry Pi 3



Figura 2. Sensor MLX90614

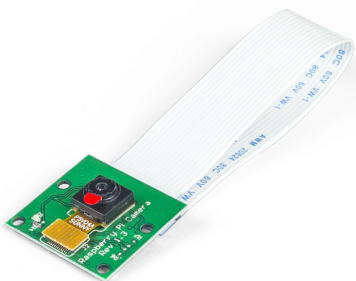


Figura 3. Módulo de câmera Rev. 1.3

- Alguns jumpers para realizar as conexões;
- Monitor comum.

E. Benefícios

Monitorar a saúde dos funcionários de um determinado estabelecimento é o principal benefício do projeto, pessoas que estiverem com sua temperatura corporal elevada e/ou sem máscara serão notificadas e deverão tomar as devidas providências que ficarão a cargo do responsável na empresa.



Figura 4. Câmera Hikvision para detecção do uso de máscara e medição da temperatura corporal

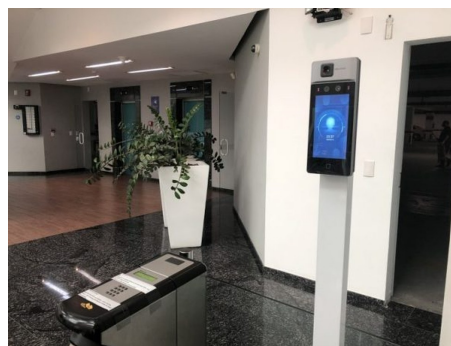


Figura 5. Câmera para detecção do uso de máscara e medição da temperatura corporal

A triagem automática pelo equipamento, eliminará a necessidade de um intermediário para a medição da temperatura e a verificação da correta utilização da máscara. Com esse processo diminuímos os riscos para as pessoas envolvidas na aferição.

F. Produtos Semelhantes

A empresa Syma Solutions localizada em Maringá, trouxe para o país a solução Hikvision, uma câmera que detecta a temperatura da pele e faz o reconhecimento facial, verificando se o indivíduo está utilizando máscara facial.

Outra empresa a GSC Segurança eletrônica, possui um modelo integrado aos acessos de catracas e portas. Enquanto faz o reconhecimento facial o produto também mede a temperatura e verifica a utilização correta da máscara pelo indivíduo.

II. DESENVOLVIMENTO DO PROJETO

Para a execução do projeto será utilizada uma Raspberry Pi 3B com o sistema operacional Ubuntu Mate instalado. A utilização desse sistema deve-se ao fato do mesmo ser leve e atender a todos os requisitos do projeto.

A Raspberry Pi estará conectada a um monitor, um teclado, um mouse e uma fonte de alimentação para a realização dos testes. O software TeamViewer será utilizado para o acesso remoto a placa.

A. Montagem Inicial

Para a montagem inicial e posterior realização de testes, utilizou-se:

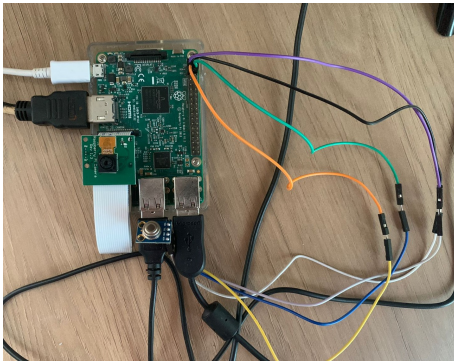


Figura 6. Montagem inicial para testes com o projeto

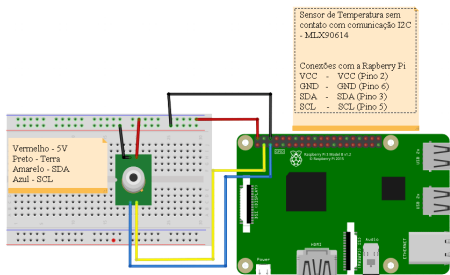


Figura 7. Montagem inicial para testes com o sensor [[3] - modificado]

- O módulo de câmera ver 1.3 para Raspberry conectado no lugar específico;
- Sensor MLX90614 conectado para a realização da comunicação I2C com a placa;
- Jumpers macho-fêmea para realizar as conexões.

B. Teste dos componentes

1) *Teste do sensor:* Primeiro deve-se ativar a conexão I2C na placa. Para isso utilizamos alguns comandos no terminal:

```
sudo raspi-config
```

Dentro da janela de configurações aberta habilita-se a comunicação I2C. Depois deve-se configurar a conexão, instalando algumas ferramentas:

```
sudo apt-get install -y i2c-tools
i2cdetect -y 1
```

Para o teste do sensor foi feito um código em linguagem C utilizando a biblioteca bcm2835.h, para a captura dos dados enviados. Os dados obtidos foram convertidos para graus celsius. Foram disponibilizados na tela para o usuário os dados da temperatura corporal e da temperatura ambiente.

O programa recebe os valores através da comunicação I2C. Converte ambas as temperaturas para graus celsius e as divide pelo número de medições realizado. Após a obtenção da temperatura corporal, essa é analisada realizando a avaliação da mesma. Caso a temperatura seja inferior a 35°C o diagnóstico é hipotermia, superior a essa, porém inferior a 37,7°C, temperatura normal e superior a última, febre.

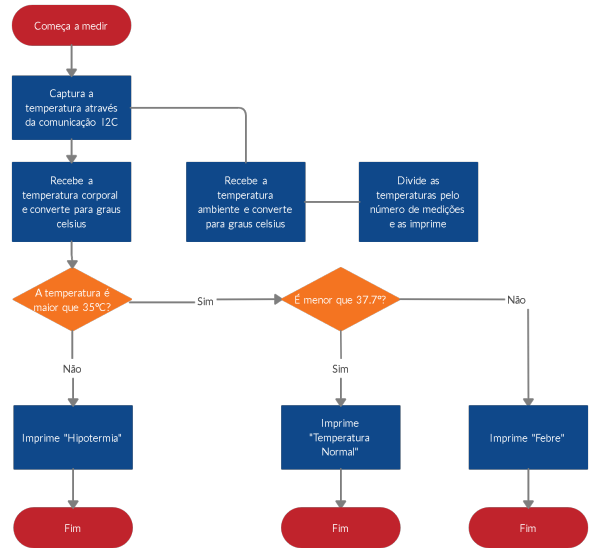


Figura 8. Fluxograma - Medição da temperatura

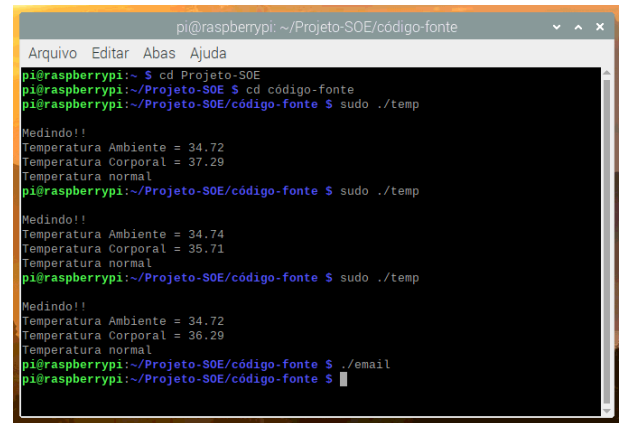


Figura 9. Medição da temperatura

2) *Teste da câmera:* Primeiro habilitou-se a câmera acendendo as configurações da Raspberry Pi através do terminal:

```
sudo raspi-config
```

Dentro da janela de configurações aberta habilitou-se o uso da câmera. Depois utilizando apenas o terminal, tiramos uma foto com o módulo, utilizando os comandos:

```
raspistill -o teste1.png -k
```

C. Instalação de bibliotecas

Para a execução do projeto algumas bibliotecas devem ser instaladas na raspberry.

1) *OpenCV:* A instalação do OpenCV foi realizada seguindo o tutorial em [5]. Foi realizado o download dos arquivos e mais alguns pacotes adicionais foram instalados.

```
sudo apt install libtiff-dev zlib1g-dev
sudo apt install libjpeg-dev libpng-dev
```

```
sudo apt install libavcodec-dev/
libavformat-dev libswscale-dev/
libv4l-dev
sudo apt-get install libxvidcore-dev/
libx264-dev
```

Para a utilização do OpenCV deve se instalar também o CMake através da linha de comando no terminal:

```
mkdir build
cd build
cmake ..
make
sudo make install
sudo ldconfig
```

2) *RaspiCam*: Para a utilização do vídeo capturado pela raspberry será utilizada a biblioteca RaspiCam instalada conforme os procedimentos em [6]. Realizou-se o download do arquivo, que foi descompactado e instalado utilizando o cmake:

```
sudo apt update
sudo apt install -y cmake
```

D. Detecção de Faces

A detecção de faces foi desenvolvida utilizando a biblioteca OpenCV e a linguagem C++. Foi utilizado o método “Haar Cascade”. Um detector baseado em “cascade”(cascata) treina uma árvore de decisão que analisa e avalia uma série de atributos em cada estágio. O sistema utiliza janelas que são classificadas em positivas e negativas. Para ser positiva essa deve passar por todos os estágios.

“Haar”é o nome dado a um dos atributos do “cascade”. São obtidos extraíndo vários pixels da imagem de acordo com os tipos de máscara escolhidos. O método “Haar cascade”foi introduzido por Viola e Jones em seu documento *Rapid Object Detection using a Boosted Cascade of Simple Features* [11], focado na detecção de rostos.

E. Detector de Máscara

O detector do uso de máscara facial foi desenvolvido para detectar primeiro a existência de um rosto na imagem. Foi utilizado o algoritmo “Haar Cascade” para identificar o rosto e a boca. O arquivo utilizado foi o padrão oferecido pela biblioteca OpenCV. Primeiro a imagem é capturada e convertida em escala de cinza e em preto e branco em seguida. As mudanças de escala facilitam o reconhecimento.

Ao não identificar um rosto em nenhuma das escalas o programa avisa que nenhum rosto foi identificado. Ao identificar apenas na escala em preto e branco o programa supõe que o usuário está utilizando uma máscara branca e, portanto confirma a utilização de máscara. Ao serem identificados um rosto em ambas as escalas o programa tenta identificar a boca. Se a boca não for identificada o usuário está utilizando uma máscara, se for identificada está sem.

A captura da imagem é feita utilizando a biblioteca RaspiCam, através da função “VideoCapture”. A imagem, variável do tipo “Mat”, é convertida para a escala de cinza e em seguida para preto e branco, para uma melhora na qualidade de detecção do haar cascade.

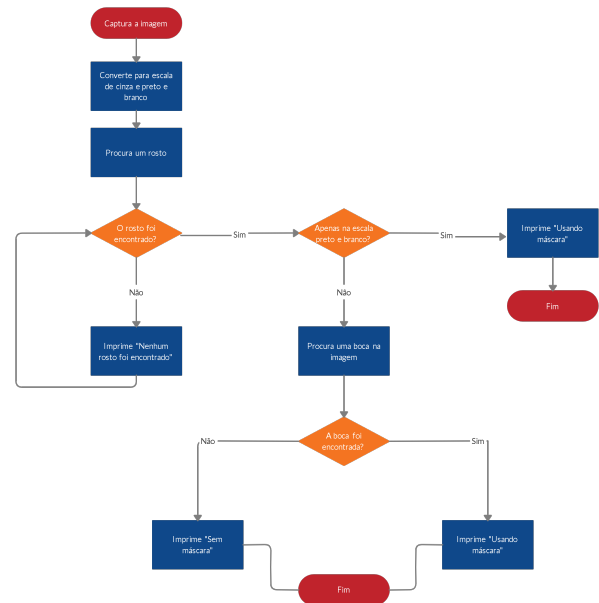


Figura 10. Fluxograma - Detecção de uso de máscara facial

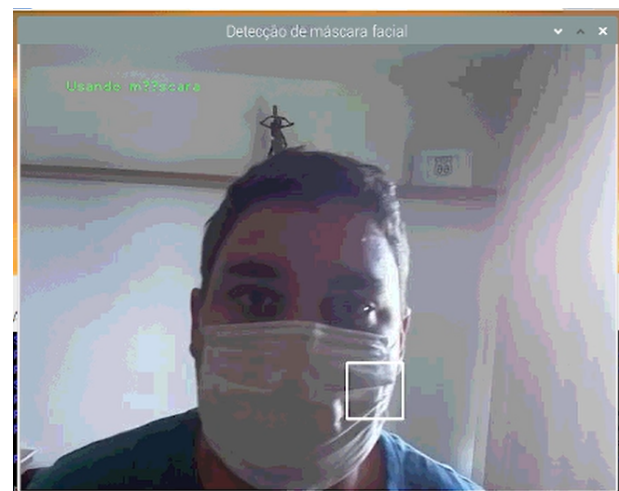


Figura 11. Rosto com a máscara detectada

F. Armazenando os dados

Para armazenar os dados utilizamos a biblioteca “stdio.h”. Foi utilizado um arquivo no formato csv facilitando a transferência de informações entre aplicativos diferentes.

Caso o arquivo não exista o arquivo é criado. “Data, Temperatura” são impressos como o cabeçalho. Depois é armazenada a temperatura medida. Caso o arquivo já exista apenas as temperaturas são armazenadas.

G. Envio de Dados por e-mail

O envio dos dados por e-mail foi realizado utilizando a biblioteca “curl.h”. Definimos o e-mail utilizado pela raspberry, quem vai receber a mensagem e os dados de um e-mail padrão como assunto e usuário.

Definimos a URL do servidor de e-mail a ser utilizado. No projeto utilizamos o Gmail. Anexamos o arquivo de dados

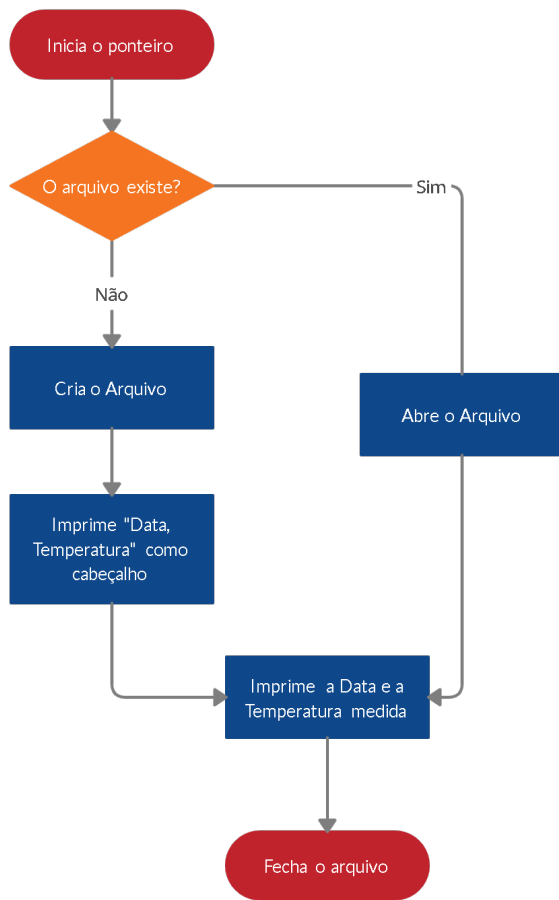


Figura 12. Fluxograma - Armazenamento de Dados

gerado anteriormente mantendo o e-mail no padrão MIME (Multipurpose Internet Mail Extensions).

O arquivo enviado pode ser visualizado na Figura 13. O arquivo CSV é visualizado no formato de tabela padrão.

A captura de tela mostra um e-mail enviado para o endereço `tortadeframboesa744@gmail.com`. O e-mail contém um anexo chamado `dados.csv`, que foi aberto e visualizado no formato de tabela. A tabela possui duas colunas: **Data** e **Temperatura**. Abaixo, apresentamos o conteúdo da tabela:

	A	B
	Data	Temperatura
1	29/10/2021 14:15	36.05
2	29/10/2021 14:15	36.31
3	29/10/2021 14:15	36.47
4	29/10/2021 15:32	37.03
5	29/10/2021 15:32	36.99
6	29/10/2021 15:32	36.17
7	29/10/2021 15:32	36.01
8	29/10/2021 15:52	37.29
9	29/10/2021 15:52	35.71
10	29/10/2021 15:52	36.29
11	29/10/2021 15:52	36.29

Figura 13. E-mail enviado pela Raspberry

H. Montagem Final

Foi construída uma pequena caixa para acomodar todos os componentes, como pode ser observado na Figura 14, com aberturas para a câmera e o sensor. Ela foi fixada na parede na altura dos olhos, conforme a Figura 15.

As medições de temperatura serão realizadas no pulso do usuário e a altura da fixação da caixa se mostrou adequada para esse propósito também. Seu design pequeno e robusto atende as especificações do projeto.

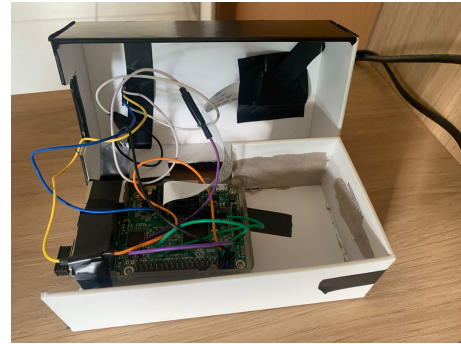


Figura 14. Acomodação dos componentes



Figura 15. Montagem Final

III. CONCLUSÃO

No início do projeto foram escolhidos os componentes necessários para implementar o termômetro infravermelho e o detector de máscaras faciais. Em seguida foram realizados testes com a câmera e com o sensor infravermelho, posteriormente foi implementado a detecção do uso da máscara e o envio dos dados coletados pelo sensor por e-mail.

Algumas dificuldades foram enfrentadas durante a confecção do projeto, como o reconhecimento do usuário, que foi abandonado depois de comprovada a sua inviabilidade. Algumas mudanças no projeto, como a adição de uma interface ou um sistema de inicialização dos códigos facilitariam a sua utilização.

Os resultados obtidos com os testes realizados podem ser considerados satisfatórios, pois tanto na detecção do uso da máscara quanto na medição da temperatura, o produto se comportou como o esperado.

REFERÊNCIAS

- [1] T. Marcela, L. Ingrid, T. Vitor, N. Mariana, A. João, C. Ricardo e N. Matheus. *VOCÊ SABE COMO SURTIU O CORONAVÍRUS SARS-COV-2?*. Disponível em: <<https://coronavirus.saude.mg.gov.br/blog/27-como-surgiu-o-coronavirus>>. Acesso 05 de ago. 2021.
- [2] Syma Informática. *Tecnologia detecta temperatura e falta de máscaras sem contato físico*. Disponível em: <<https://g1.globo.com/pr/norte-noroeste/especial-publicitario/syma-informatica/noticia/2020/06/24/tecnologia-detecta-temperatura-e-falta-de-mascaras-sem-contato-fisico.ghtml>>. Acesso 05 de ago. 2021.
- [3] NSC TV. *Coronavírus : Tecnologia detecta temperatura e falta de máscaras sem contato físico*. Disponível em: <<https://g1.globo.com/sc/santa-catarina/techsc/noticia/2020/07/12/coronavirus-tecnologia-detecta-temperatura-e-falta-de-mascaras-sem-contato-fisico.ghtml>>. Acesso 05 de ago. 2021.
- [4] U. Taher. *IoT Based Contactless Body Temperature Monitoring using Raspberry Pi with Camera and Email Alert*. Disponível em: <<https://circuitdigest.com/microcontroller-projects/iot-based-contactless-body-temperature-monitoring-using-raspberry-pi-with-camera-and-email-alert>>. Acesso em 01 de set. de 2021.
- [5] Arduino e cia. *Como usar a Câmera na Raspberry Pi*. Disponível em: <<https://www.arduinoecia.com.br/como-usar-camera-raspberry-pi/>>. Acesso em 01 de set. de 2021.
- [6] Solarian Programmer *Install OpenCV 4 on Raspberry Pi for C++ and Python development*. Disponível em: <<https://solarianprogrammer.com/2019/09/17/install-opencv-raspberry-pi-raspbian-cpp-python-development/>>. Acesso em 08 de out. 2021.
- [7] AVA *RaspiCam: C++ API for using Raspberry camera with/without OpenCv*. Disponível em: <<https://www.uco.es/investigacion/grupos/ava/node/40>>. Acesso em 08 de out. 2021.
- [8] Lindevs *Install CMake on Raspberry Pi*. Disponível em: <<https://lindevs.com/install-cmake-on-raspberry-pi/>>. Acesso em 15 de out. 2021.
- [9] S. Saurabh. *Face Mask Detection: Simple OpenCV Based Program*. Disponível em: <<https://medium.com/@saurabh.shaligram/face-mask-detection-simple-opencv-based-program-417bbcf0abd8>>. Acesso em 15 de out. 2021.
- [10] T. Arjun. *Opencv C++ Tutorial of Face(object) Detection Using Haar Cascade* Disponível em: <<http://opencv-tutorials-hub.blogspot.com/2016/03/how-to-do-face-detection-opencv-haar-cascade.html>>. Acesso em 18 de out. 2021.
- [11] Paul V., Michael J. *Rapid Object Detection using a Boosted Cascade of Simple Features*. Disponível em: <<https://www.cs.cmu.edu/efros/courses/LBMV07/Papers/viola-cvpr-01.pdf>>. Acesso em 18 de out. 2021.
- [12] Adrian R. *OpenCV Haar Cascades*. Disponível em: <<https://www.pyimagesearch.com/2021/04/12/opencv-haar-cascades/>>. Acesso em 22 de out. 2021.