

Rede Neural Convolutacional para a Classificação de Imagens de Pessoas Fazendo Uso de Máscaras do Tipo EPI

Pedro Henrile Salvador¹, André Luis ALbuquerque Pinheiro¹, Francisco Cleber da Conceição Feitosa², Robson Gonçalves Fechine Feitosa¹

¹Instituto Federal de Educação, Ciência e Tecnologia do Ceará – Campus Crato (IFCE)
Rodovia CE 292, KM 15, Gisélia Pinheiro - CEP 63115-500 - Crato - CE – Brazil

²Departamento de Engenharia Elétrica - Universidade Federal do Piauí – Brazil (UFPI)

{pedro.henrile.salvador07, andre.luis.albuquerque07}@aluno.ifce.edu.br,
cleberfeitosa.ufpi@gmail.com, robsonfeitosa@ifce.edu.br

Abstract. *The present work used Computer Vision and Convolutional Neural Network (CNN) techniques, with the aim of classifying images of people using or not using PPE (Personal Protective Equipment) masks. To this end, an image repository was sought for training, validation and testing; LabelImg was used to label the training images, and CNN was used to build the model. After several experiments, the best results obtained were Accuracy of 1.0, Coverage of 0.88 and F-Measure of 0.93.*

Resumo. *O presente trabalho utilizou-se de técnicas de Visão Computacional e Rede Neural Convolutacional (CNN), com o objetivo de classificar imagens de pessoas fazendo uso ou não de máscaras do tipo EPI (Equipamento de Proteção Individual). Para tanto, buscou-se um repositório de imagens para treinamento, validação e testes; utilizou-se o LabelImg para a rotulação das imagens de treinamento, e CNN para a construção do modelo. Após diversos experimentos, os melhores resultados obtidos foram Precisão de 1.0, Cobertura de 0.88 e Medida-F de 0.93.*

1. Introdução

As aplicações que utilizam Visão Computacional (VC), Ciência de Dados (CD) e Inteligência Computacional (IC), de forma conjunta ou separadamente, têm ganhado espaço nas mais diversas áreas, como: carros autônomos, reconhecimento biométrico, computação afetiva, monitorização de ambientes, processamento de imagens de satélites e *games*. O termo 4.0 se popularizou, tornando-se uma espécie de sufixo para designar indústrias, cidades e agronegócios que se utilizam das tecnologias supracitadas para modernização de seus serviços e produtos.

Desde o surgimento da COVID-19, uma infecção respiratória aguda causada pelo coronavírus SARS-CoV-2, em 2019 na China, de acordo com a Organização Mundial da Saúde (OMS) [Who 2024] foram notificados mais de 775 milhões de casos de pessoas que contraíram a doença, e mais de 7 milhões chegaram a óbito, em todo o mundo segundo último registro do dia 22 de junho de 2024. Dentre as ações de combate à pandemia, a utilização de máscaras de proteção facial do tipo EPI (Equipamento de Proteção Individual) tornou-se um dos mecanismos mais eficientes para prevenção da disseminação do vírus, pois evita que as pessoas toquem o rosto com as mãos; reduz o

risco de infecção nas proximidades de uma pessoa infectada; e, reduz a inalação do vírus mesmo na presença de pessoas infectadas [Kähler and Hain 2020].

Desta forma, políticas públicas têm sido empregadas tornando obrigatório o uso de tais máscaras em ambientes críticos fechados, onde existe grande circulação de pessoas, como transportes coletivos, e ambientes de tratamento de saúde, como hospitais, clínicas e Unidades de Pronto Atendimento (UPAs) [Floriano et al. 2024]. No entanto, apesar da criação de novas regras, leis e decretos sobre a obrigatoriedade do uso de máscaras em ambientes fechados, na prática é muito difícil e financeiramente oneroso garantir o seu cumprimento. Logo, a utilização de ferramentas automatizadas de monitoramento de utilização de máscaras, podem apoiar tal processo. Na literatura científica, diferentes abordagens que fazem uso de VC, CD e IC têm surgido com o intuito de oferecer esse tipo de apoio, conforme ilustrado nas próximas seções.

Logo, o presente trabalho propõe uma abordagem para a classificação automatizada de indivíduos que utilizam máscaras de proteção, com o auxílio de técnicas de VC, IC e processamento de imagem. Inicialmente, realizou-se uma revisão da literatura e pesquisa por repositórios de imagens. Em seguida, foram conduzidos os procedimentos de tratamento das imagens para treinamento e validação dos mecanismos propostos. Através da presente abordagem, espera-se contribuir no auxílio ao desenvolvimento de soluções tecnológicas eficazes para o monitoramento e controle da disseminação de doenças infecciosas. Para isso, o presente trabalho foi estruturado da seguinte forma: na Seção 2 são apresentados os trabalhos relacionados; na Seção 3 são apresentados os materiais e métodos utilizados; os resultados são discutidos na Seção 4; e, por fim, na Seção 5, as conclusões são dispostas, assim como são demarcados os trabalhos futuros.

2. Trabalhos Relacionados

A classificação automatizada de indivíduos utilizando máscara de proteção tem sido um campo de crescente interesse, especialmente com o advento da pandemia do COVID-19. Diversos estudos têm explorado soluções eficazes nesse contexto. [Lodh et al. 2020] desenvolveram um sistema para realizar a identificação de máscara facial no mundo real, por meio de uma Rede Neural Artificial. A aplicação também realizou o reconhecimento de pessoas, caso a mesma não apresente a máscara de proteção, e a notificação através de uma mensagem por e-mail. Nela, são detectados os rostos e máscaras faciais por vários ângulos em um curto período de tempo. Foram utilizadas imagens da plataforma Kaggle e do Google Images obtidas com o auxílio de um script em Python. No trabalho, é utilizado o MobileNetV2 para a geração do modelo de treinamento e o Scikit-Learn para converter em binário as marcas de classe, dividir o conjunto de dados e imprimir o relatório de classificação. Segundo os autores, a aplicação apresentou resultados satisfatórios e conseguiu classificar com sucesso pessoas com e sem máscara, além de identificar e notificar quem estava sem máscara.

[Balasundaram et al. 2021] apresentaram um sistema para identificação de pessoas utilizando ou não máscara de proteção em locais públicos, além disso, os administradores do sistema são notificados por e-mail, caso haja alguém sem máscara de proteção. Foi utilizado o TensorFlow para a construção do modelo de treinamento, e os pesos dos modelos de detecção de objetos baseados no MobileNetV2 para realizar o treinamento do modelo. Foi utilizada a função *Image Data Generator* para o aumento

dos dados, no entanto não foram detalhadas informações sobre a base de treinamento. O modelo apresentou 90% de precisão, mas o trabalho não detalhou maiores informações referentes aos testes realizados para validação do modelo.

[Suganthalakshmi et al. 2021] implementaram um modelo de algoritmo para detecção em evocação e não cobertura, que verifica e alerta se a pessoa no vídeo está utilizando máscara ou não, onde, segundo os autores, o objetivo do trabalho é evitar/diminuir a proliferação da COVID-19 e futuras outras doenças que são facilmente transmissíveis pelo ar, além disso, o trabalho pode servir como base para outras pesquisas, como aprendizado de detecção de objetos no geral. Para a extração dos conjuntos de dados (pessoas com máscara e sem máscaras) foi utilizado o MobileNetV2, e como modelo uma *Convolutional Neural Network* (CNN) em conjunto com as ferramentas OpenCV e Keras. No trabalho não foi apresentado o resultado da acurácia do algoritmo, contudo, o trabalho se mostra promissor devido sua integração com recursos de câmeras que podem ser implementados em caixas eletrônicos, bancos, escolas, dentre outros ambientes, e por sua metodologia simples e prática para detecção de máscaras.

[Vrigkas et al. 2022] apresentaram a criação da FaceMask, uma base de imagens contendo pessoas com e sem máscara de proteção. A base conta com 4866 imagens obtidas por meio do Google Images, e foi disponibilizada publicamente para facilitar a realização de trabalhos futuros. Para validação da base foram analisadas algumas versões do YOLO: YOLOv3, YOLOv4 e YOLOv4-tiny. Onde, a versão YOLOv4 apresentou o melhor desempenho, quando comparada com as demais, apresentando os seguintes resultados: Mean Average Precision (mAp) 90.04, Average Intersection over Union (IoU) 68.99, AP Mask 89.98 e AP No Mask 90.10. [Shorten and Khoshgoftaar 2019] se utiliza de *Data Augmentation*, com ênfase no aprendizado profundo. Para isso, a técnica realiza ajustes por meio de rotações da imagem em diversos ângulos (horizontal e vertical), filtros de cores e ruídos, e conseqüentemente aumentando os dados do conjunto de treinamento.

A Tabela 1 sumariza as principais características dos trabalhos apresentados nesta seção. É possível observar que todos os trabalhos utilizaram a ferramenta OpenCV e CNN para manipulação e processamento de imagens, e treinamento do modelo, uma vez que o YOLO também faz uso do CNN; e, quatro trabalhos utilizaram o MobileNet. Apenas dois trabalhos apresentaram o detalhamento do desempenho do sistema por meio de métricas de avaliação, como Precisão e Cobertura. Os trabalhos que indicaram resultados qualitativos citaram que seus sistemas apresentaram bom desempenho, sem maiores detalhes de métricas. Também é possível observar que todos os trabalhos utilizaram bases de imagens públicas.

Tabela 1. Ferramentas e métodos dos trabalhos encontrados na literatura

Trabalho	Ferramentas	Modelo	Resultados	Base
[Lodh et al. 2020]	Scikit-Learn, Mobile-NetV2, OpenCV.	CNN	Qualitativo	Púb.
[Balasundaram et al. 2021]	TensorFlow, MobileNetV2, OpenCV, Image Data Generator.	CNN	Qualitativo	Púb.

[Vrigkas et al. 2022]	OpenCV, LabelImg, Darknet.	YOLOv3, v4, v4-tiny	mAp: 90.04 IoU: 68.99 Mask: 89.98 NoMask: 90.10	Púb.
[Suganthalakshmi et al. 2021]	OpenCV, TensorFlow, MobileNetV2.	CNN	Qualitativo	Púb.
Proposta deste trabalho	MobileNetV2, OpenCV, Labe-Img.	CNN	Precisão: 1, Cobertura.: 0.88, Medida-F: 0.93	Púb.

Vale ressaltar que a maioria dos trabalhos mencionados detalha suas técnicas metodológicas, mas frequentemente omite informações cruciais sobre a implementação dos algoritmos e o desempenho das métricas alcançadas. Embora o sigilo dos dados possa ser necessário em alguns casos, é essencial que haja uma base acessível para desenvolvedores e pesquisadores compreenderem como esses estudos foram conduzidos.

3. Materiais e Método

O desenvolvimento do presente trabalho foi dividido em etapas, que envolvem: 1) análise da literatura; 2) busca de imagens representando a face de pessoas com e sem utilização de máscara do tipo EPI; 3) processamento das imagens; 4) rotulação das imagens de treinamento com LabelImg; 5) treinamento do modelo R-CNN para a classificação das imagens quanto a presença ou ausência de utilização de máscara; e, 6) validação do modelo por meio de métricas de Precisão, Cobertura e Medida-F.

A análise da literatura permitiu observar os principais métodos, modelos, ferramentas e bases de imagens utilizadas. Assim, foi possível identificar trabalhos que utilizaram imagens públicas disponibilizadas na Web, bem como, base de dados em repositórios como a plataforma Kaggle. Logo, no presente trabalho utilizou-se duas bases de imagens. Uma base para construção do treinamento do modelo R-CNN, e outra base para validação do modelo já treinado. A base de treinamento consistiu de imagens públicas disponibilizadas na plataforma Kaggle na base Face Mask Detection¹. Para a base de validação foi construída uma base própria com 101 imagem de dimensões de 640x480 pixels, a partir de fotos capturadas com uma câmera Logi 720p². As imagens foram capturadas sob diferentes condições de iluminação e diferentes horários do dia, refletindo iluminação natural e artificial, disponibilizada no repositório CNN-Mascara-EPI³.

Para a etapa de treinamento, foi utilizado o notebook Acer Aspire com processador Intel Core i5 10210U, 8GB de memória RAM DDR4 e placa gráfica NVIDIA GeForce MX250 com 2GB de memória dedicada GDDR5, onde a utilização

¹Link para o repositório das imagens em:

<https://www.kaggle.com/datasets/andrewmvd/face-mask-detection>. Último acesso em: 08 de jun. de 2024.

²Link para consultar especificações da câmera Logi 720p:

<https://www.logitech.com/pt-br/products/webcams/c270-hd-webcam.960-000694.html>. Último acesso em: 09 de jun. de 2024.

³Link para o repositório com o código-fonte e a base de imagens:

<https://github.com/AndreAlbu/CNN-Mascaras-EPI>. Último acesso em: 10 de ago de 2024.

da GPU favoreceu o processo de treinamento do modelo; e, conforme detalhado na Tabela 2, o processo de treinamento do modelo demandou o tempo médio de 1:36 horas.

Ainda é importante mencionar que foi utilizada a linguagem de programação Python e a biblioteca *Open Source TensorFlow* para suporte no processo de treinamento e validação do modelo [Goldsborough 2016; Gavai et al. 2017], e como arquitetura da rede utilizou-se o MobileNet [Jaime 2020]. Para o processamento e visualização das imagens foi utilizada a biblioteca OpenCV, combinada com a biblioteca Matplotlib para a geração de gráficos [Ari and Ustazhanov 2014].

3.1. Processamento das imagens

Devido ao número limitado de imagens encontradas que satisfizesse os critérios: (i) imagens de pessoas com máscara, onde na imagem deve-se haver pelo menos uma pessoa com máscara; (ii) pessoas sem uso de máscara, onde deve-se haver somente pessoas sem máscara. Utilizou-se de técnicas de processamento de imagem para aumentar a quantidade de imagens da base. Foi utilizado o processo de *Data Augmentation*, a técnica de aumento de dados permite criar variações das imagens originais, enriquecendo a base e promovendo maior robustez e generalização dos modelos [Corrêa, 2023]. A Figura 1, ilustra os processos realizados nas imagens.

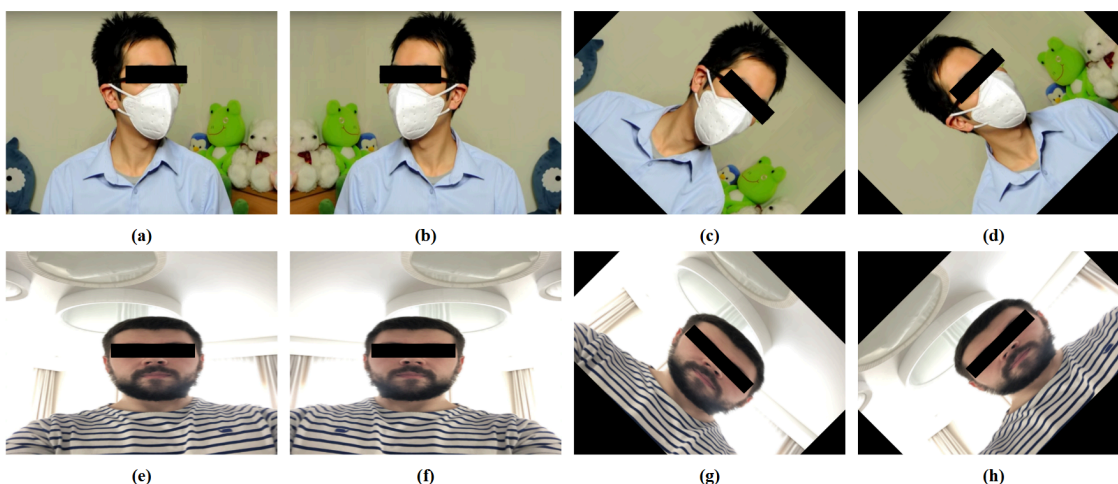


Figura 1. Processo de ajuste das imagens utilizadas para realizar o treinamento da CNN: imagens ajustadas (a) com máscara e (e) sem máscara; imagens transladadas (b) com máscara e (f) sem máscara; imagens rotacionadas para direita (c) com máscara e (g) sem máscara; imagens rotacionadas para esquerda (d) com máscara e (h) sem máscara.

Para a construção da base “Ajustadas”, houve um ajuste da largura e altura das imagens, por meio da função *resize()* da biblioteca do OpenCV. A base “Translação” foi construída a partir das mesmas imagens da base anterior utilizando a função *flip()* do OpenCV. Por fim, a base “Rotação” utilizou-se da função do OpenCV *warpAffine()*, para rotacionar as imagens da base anterior em 45° para a esquerda, e 45° para a direita. O código-fonte do processamento das imagens está disponibilizado no repositório CNN-Mascara-EPI.

3.2. LabelImg

Uma importante etapa do presente trabalho consiste na rotulação manual das imagens de treinamento, por meio da ferramenta LabelImg⁴. A Figura 2 ilustra tal processo, onde, uma vez realizada a marcação, os metadados associados à imagem rotulada são utilizados no processo de treinamento junto à biblioteca TensorFlow. Ela permite realizar anotações em imagens, a fim de localizar o objeto de interesse na imagem. As anotações podem ser salvas em XML no formato PASCAL VOC, YOLO e CreateML. Para o presente trabalho foram utilizadas apenas as anotações em PASCAL VOC, contendo detalhes como: a caixa delimitadora (localização do objeto dentro da imagem), classes e rotação. Ela permite rotular mais de uma classe na imagem, no entanto cada imagem possui uma anotação única referente a todas as classes.



Figura 2. Processo de marcação das imagens utilizando o LabelImg.

3.3. Treinamento da R-CNN

No processo de treinamento supervisionado, as imagens foram rotuladas utilizando o LabelImg, onde foram definidas duas classes, com-máscara e sem-máscara. Assim, foram realizadas anotações na região da face das pessoas que estavam presentes na imagem, onde, foi considerado que as pessoas classificadas como “com-máscaras” eram aquelas que possuíam qualquer tipo de máscara na face independente da cor da máscara. Já as imagens das pessoas que não possuíam máscara na região da face ou a máscara estava mal posicionada (não cobrindo nariz e boca) eram classificadas como “sem-máscara”.

Após a supervisão das imagens, os arquivos gerados foram convertidos através de um script Python para o formato TFRecord, que consiste em uma sequência de registros binários [TensorFlow 2022]. Foi utilizado o modelo pré-treinado do MobileNetV2 como arquitetura padrão da rede, e o pipeline de configuração do COCO17⁵, e por fim, a rede foi treinada em 5000 épocas.

⁴Link para download da ferramenta LabelImg: <https://github.com/tzutalin/labelImg>. Último acesso em: 08 de jun de 2024.

⁵Link para download do modelo COCO17: http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8.tar.gz. Último acesso em: 08 de jun de 2024.

Para o presente trabalho foram realizados três treinamentos, o primeiro treinamento foi somente para a base de imagens Ajustadas. O segundo treinamento, englobou a base Ajustadas e a base de imagens Translação. Por fim, o último treinamento compreendeu as bases, Ajustadas, Translação e a base Rotação. Para o treinamento de cada modelo, as imagens foram divididas em 70% para *train* e 30% para *test* [UBAI, 2023]. A Tabela 2 ilustra o quantitativo de imagens, bem como as suas dimensões para a etapa de treinamento, além do tempo demandado para seu processamento.

Tabela 2. Detalhamento das bases de imagens.

Treinamento	Base	Train	Test	Total	Dimensões(px)	Tempo
1	Ajustadas	79	56	135	640x480	1:28 horas
2	Translação	189	81	270	640x480	1:36 horas
3	Rotação	378	162	540	640x480	1:44 horas

Foi utilizado o modelo de configuração COCO17 uma vez que ele simplifica o processo de treinamento da rede, bastando realizar somente o treinamento da rede com as devidas marcações nas imagens conforme discutido na seção anterior. O processo de implementação dos treinamentos da R-CNN foram realizados com apoio da biblioteca TensorFlow, onde foram utilizados os parâmetros padrões da biblioteca para configuração do número de camadas, função de ativação, dentre outros, conforme código-fonte disponibilizado no repositório CNN-Mascara-EPI.

4. Resultado e Discussão

A avaliação do desempenho de modelos de classificação é crucial para validar a eficácia das técnicas propostas. [Schütze et al. 2008] apresenta diversas métricas para avaliar o desempenho dos sistemas de recuperação de informação. Dentre as métricas citadas, os autores definem “Precisão” como a fração de documentos recuperados que são relevantes; “Cobertura” como a fração de documentos relevantes que são recuperados; e, “Medida-F” como a média harmônica entre os pesos da “Precisão” e “Cobertura”. Assim, tais métricas estão formalizadas conforme as equações 1, 2 e 3 respectivamente, e detalhadas a seguir.

$$(1) \text{ Cobertura} = \frac{VP}{VP + FN}$$

$$(2) \text{ Precisão} = \frac{VP}{VP + FP}$$

$$(3) \text{ Medida-F} = \frac{\text{Precisão} * \text{Cobertura}}{\text{Precisão} + \text{Cobertura}}$$

No presente trabalho, adotou-se que: os documentos relevantes são todas as imagens da base que devem ser classificadas pelo modelo; os documentos relevantes não recuperados são todas as imagens onde o modelo não retornou uma classificação. Logo, temos que: Verdadeiros Positivos (VP) correspondem ao total de imagens

corretamente classificadas; Falsos Positivos (FP) correspondem ao total de imagens incorretamente classificadas; Falsos Negativos (FN) e Verdadeiros Negativos (VN) correspondem às imagens que o modelo não retornou classificação.

Foi realizada uma análise das imagens contendo a face de pessoas fazendo uso ou não de máscaras, sob diferentes angulações, i.e., fotos frontais e de perfil. Conforme ilustrado na Figura 3 nos itens (a), (b), (c) e (d).

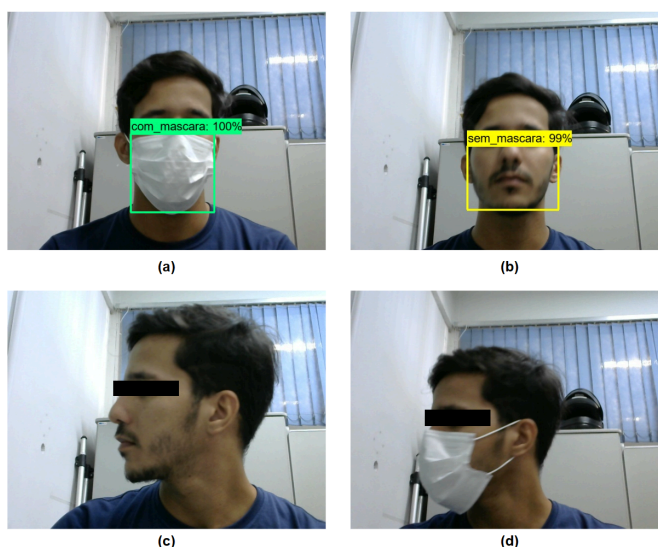


Figura 3. Imagens da base de validação, utilizada para avaliação do modelo: (a) identificação correta do uso de máscara; (b) identificação correta do não uso de máscara, (c) identificação incorreta do uso de máscara; (d) identificação incorreta do uso de máscara.

A Tabela 3 ilustra o desempenho do sistema na base de imagens de validação, com destaque para o valor médio de 0.89 para Medida-F.

Tabela 3. Resultados da validação utilizando a câmera Logi 720p.

Treinamento	Base	Imagem	Precisão	Cobertura	Medida-F
1	Ajustadas	Face de Perfil	1.0	0.77	0.87
	Ajustadas	Face de Frente	1.0	0.97	0.98
2	Translação	Face de Perfil	1.0	0.80	0.89
	Translação	Face de Frente	1.0	0.97	0.98
3	Rotação	Face de Perfil	0.90	0.70	0.79
	Rotação	Face de Frente	0.90	0.78	0.84

Após a avaliação das dos modelos treinados, obteve-se o melhor resultado no treinamento 2 com resultado: precisão média de 1.0, cobertura média de 0.88, e medida-f com média de 0.93. O treinamento 1 apresentou o segundo melhor resultado com: precisão média de 1.0, cobertura média de 0.87, e medida-f com média de 0.92. Por fim, o treinamento 3 apresentou o pior desempenho, mesmo com a quantidade de

imagens sendo superior aos outros treinamentos, o que pode indicar que ocorreu um *overfitting* [Ying, 2019]. O treinamento 3 apresentou os resultados: precisão média de 0.90, cobertura média de 0.74, e medida-f com média de 0.81.

Observou-se que imagens com a face de perfil resultaram em um desempenho inferior, quando comparado com imagens frontais, influenciando negativamente o desempenho do algoritmo nesses casos. Nessas imagens, para ambos os lados (direito e esquerdo) da face, o algoritmo não foi capaz de retornar classificações corretas. Outro ponto importante a ser destacado é que, em alguns casos, com imagens da face de perfil com o uso da máscara, conforme ilustrado na Figura 3, item (c) e (d), o algoritmo não retorna classificação nenhuma pois não conseguiu identificar se estava com ou sem máscara devido ao ângulo de face de perfil. Para imagens com a face de perfil, sem a utilização da máscara, o algoritmo não retornou a saída, i.e., não realizou a classificação, o que considerou-se como FN ou VN. Em todas as imagens sem máscara, o algoritmo retornou a classificação corretamente. Por fim, utilizando o melhor modelo treinado para realizar a classificação em tempo-real, por meio de vídeo, é possível observar uma média de 1 a 2 segundos para que ele forneça a resposta, dada a mesma configuração de computador utilizada no treinamento do modelo.

5. Conclusão

A utilização de máscaras de proteção facial do tipo EPI tornou-se um dos mecanismos mais eficientes para prevenção da disseminação da COVID-19, contudo, verificar e garantir o cumprimento da utilização de máscaras é um processo manualmente difícil. Logo, ferramentas computacionais automatizadas de monitoramento de utilização de máscaras podem apoiar tal processo.

Assim, o presente trabalho detalhou o processo de busca e utilização de repositórios para treinamento e validação de um modelo de classificação de imagens de pessoas fazendo uso ou não de máscaras do tipo EPI. Também foram apresentadas as principais ferramentas computacionais necessárias para realizar tal processo. Para avaliar o modelo aqui apresentado, foram realizados diversos experimentos, sob diferentes ajustes nas bases de imagens onde, como principal resultado, obteve-se Precisão de 1, Cobertura de 0.88 e Medida-F de 0.93.

Como trabalhos futuros, espera-se analisar o desempenho do presente modelo incorporado a um sistema de videomonitoramento, em tempo real, de ambientes como hospitais e clínicas, onde a utilização do EPI é fundamental na prevenção de doenças.

Referências

- Ari, N. and Ustazhanov, M. (2014). Matplotlib in python. In *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*, pages 1–6. IEEE.
- Balasundaram, A., Kumar, N., Sivaraman, A. K., Vincent, R., and Rajesh, M. (2021). Mask detection in crowded environment using machine learning. In *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, pages 1202–1206. IEEE.
- Floriano, I., Silvinato, A., Bacha, H. A., Barbosa, A. N., Tanni, S., & Bernardo, W. M. (2024). *Eficiência do uso de máscaras durante o surto de COVID-19 em estudos de*

coorte e caso-controle: uma revisão sistemática e meta-análise. *Jornal Brasileiro de Pneumologia*, 49, e20230003.

Gavai, N. R., Jakhade, Y. A., Tribhuvan, S. A., and Bhattad, R. (2017). Mobilenets for flower classification using tensorflow. In **2017 international conference on big data, IoT and data science (BID)**, pages 154–158. IEEE.

Goldsborough, P. (2016). *A tour of tensorflow*. *arXiv preprint arXiv:1610.01178*.

Jaime, T. F. (2020). Uso de algoritmos de aprendizado de máquina supervisionado para rotulação de dados.

Kähler, C. J., and Hain, R. (2020). Fundamental protective mechanisms of face masks against droplet infections. *Journal of aerosol science*, 148, 105617. <https://doi.org/10.1016/j.jaerosci.2020.105617>

Lodh, A., Saxena, U., Motwani, A., Shakkeera, L., Sharmasth, V. Y., et al. (2020). Prototype for integration of face mask detection and person identification model–covid-19. In **2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)**, pages 1361–1367. IEEE.

Schutze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Corrêa, G. P., Colombini, E. L., Técnico-IC-PFG, R., & de Graduação, P. F. (2023). *Aumento de dados com modelos de difusão image to image e GANs para melhoria na generalização de detectores de deepfake*.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Suganthalakshmi, R., Hafeeza, A., Abinaya, P., and Devi, A. G. (2021). *Covid-19 face-mask detection with deep learning and computer vision*. *Int. J. Eng. Res. Tech.(IJERT) ICRADL*.

UBAI(2023). *Best data augmentation techniques [2024 update]*. Disponível em: <https://ubiai.tools/what-are-the-advantages-anddisadvantages-of-data-augmentation-2023-update>. Acesso em: 9 jul. 2024.

TensorFlow (2022). Load and preprocess data. Acessado: 2024-04-05.

Vrigkas, M., Kourfalidou, E.-A., Plissiti, M. E., and Nikou, C. (2022). *Facemask: A new image dataset for the automated identification of people wearing masks in the wild*. *Sensors*, 22(3):896.

Who, W. H. O. (2024). *Coronavirus disease (covid-19) outbreak situation*. Disponível em: <https://www.who.int/emergencies/diseases/novel-coronavirus-2019>. Acesso em: 09 jul. 2024.

Ying, X. (2019, February). *An overview of overfitting and its solutions*. In *Journal of physics: Conference series* (Vol. 1168, p. 022022). IOP Publishing.