ISSN 1870-4069

Dataset para la detección de elementos de bioseguridad facial mediante técnicas de aprendizaje computacional

Carlos Vicente Niño Rondón, Diego Andrés Castellano Carvajal,
Sergio Alexander Castro Casadiego, Byron Medina Delgado, Dinael Guevara Ibarra

Universidad Francisco de Paula Santander,
Colombia

{carlosvicentenr,diegoandrescc,sergio.castroc,byronmedina,
dinaelgi}@ufps.edu.co

Resumen. En este documento se desarrolla la prueba a un dataset elaborado con tomas de imágenes y videos en la zona céntrica de la ciudad de Cúcuta, con el fin de determinar si las personas que circulan por espacios comerciales portan o no el tapabocas como elemento de bioseguridad facial. El dataset se obtuvo de 1450 imágenes inicialmente capturadas y que, mediante técnicas de aumentado de datos por variaciones en la forma de las imágenes, se aumentaron a 14067 imágenes. Las técnicas de aprendizaje computacional utilizadas son clasificadores en cascada y redes neuronales convolucionales, ambas aplicadas en lenguaje de programación Python. Con el sistema de clasificador en cascada se obtuvo un acierto en las detecciones de 80.17 %, mientras que con la red neuronal convolucional desarrollada se obtuvo un acierto de 90.19 %. Teniendo en cuenta la estructura de las técnicas de aprendizaje, así como las tasas de acierto, se infiere la fiabilidad del conjunto de datos obtenido a la hora de determinar qué personas portan elementos de bioseguridad facial en espacios comerciales.

Palabras clave: dataset, imágenes, clasificador en cascada, red neuronal convolucional.

Dataset for Detection of Facial Biosafety Elements Using Computational Learning Techniques

Abstract. This document develops the test to a dataset elaborated with images and videos taken in the downtown area of the city of Cúcuta, in order to determine whether or not people circulating in commercial spaces wear face masks as an element of facial biosecurity. The dataset was obtained from 1450 images initially captured and, by means of data augmentation techniques due to variations in the shape of the images, was increased to 14067 images. The computational learning techniques used are cascade classifiers and convolutional neural networks, both implemented in Python programming language. With the cascade classifier system, a detection accuracy of 80.17 % was obtained, while with the convolutional neural network developed, an accuracy of 90.19 % was

obtained. Taking into account the structure of the learning techniques, as well as the accuracy rates, the reliability of the data set obtained can be inferred when determining which people are wearing facial biosecurity elements in commercial spaces.

Keywords: dataset, images, cascade classifier, convolutional neural network.

1. Introducción

Los elementos de bioseguridad hacen referencia al conjunto de dispositivos y herramientas utilizados para mitigar las posibilidades de contagio y los efectos generados por diversos agentes biológicos. Durante la pandemia de la COVID-19, los estados y comunidades académicas y científicas han recomendado el uso de elementos de bioseguridad facial, como método para frenar la transmisión del virus entre humanos, que mínimamente se produce con gotas respiratorias de hasta 5 micras en distancias de hasta dos metros [1]. Así mismo, la pandemia trajo consigo estragos en los sectores económicos y productivos de los países, por lo que mediante estrategias conjuntas se ha iniciado con la reactivación económica en los mismos, sin dejar de lado los cuidados mínimos para evitar el aumento acelerado en el número de contagios.

Las técnicas de aprendizaje computacional estudian los algoritmos que generan procesos de aprendizaje mediante guías o series de ejemplos [2]. Además, mediante técnicas de visión por computadora y los métodos de aprendizaje automático o aprendizaje profundo, se adquieren y analizan imágenes del mundo real y se produce información tratable en una computadora [3]. Entre las técnicas de aprendizaje automático para el tratamiento de imágenes destacan los detectores en cascada, donde mediante clasificadores basados en funciones generalmente tipo Haar [4], y en el entrenamiento de imágenes positivas, (correspondientes a imágenes donde se encuentre el objeto en cuestión), e imágenes negativas, (donde no se encuentre el objeto a detectar), se realiza la detección de los mismos, en imágenes que el sistema no ha visto [5]. Por otra parte, entre las técnicas de aprendizaje profundo destacan las redes neuronales convolucionales, donde mediante múltiples capas de filtros de convolución, inicialmente se extraen las características de las imágenes, se reducen por muestreo y se obtienen neuronas simples para ejecutar la clasificación, según las características extraídas [6].

Para aplicar los métodos y procesos de visión y aprendizaje computacional, se requiere de un conjunto de imágenes en las que se presenten las características de el/los objetos a detectar [7]. Dicho conjunto de imágenes o dataset, debe contener un número de imágenes que se encuentran en las escalas de los miles, por lo que, en ocasiones, se requieren de técnicas de aumentado de datos, donde por modificaciones leves de forma en las imágenes originales, se logra la expansión en el número de imágenes para entrenamiento y pruebas [8].

En este documento se presentan las pruebas a un dataset elaborado con imágenes capturadas en espacios comerciales de la ciudad de Cúcuta, Colombia, y completado mediante técnicas de aumentado de datos. El dataset, se prueba mediante un sistema de

clasificadores en cascada y una red neuronal convolucional, ponderando los aciertos en las detecciones en cada uno de los procesos. La adquisición de las imágenes se realiza mediante una cámara de 13 megapíxeles, mientras que la codificación se realiza en lenguaje Python, con herramientas de sistema operativo Windows.

2. Métodos

La metodología propuesta consta de 3 etapas. En la primera etapa se realiza la adquisición de la imagen y se consideran los factores requeridos para la obtención del dataset inicial, como el nivel de luminosidad, hora de la adquisición, altura a la que se ubica el dispositivo de captura de imágenes y videos, y los formatos en los que obtienen las tomas. Asimismo, para la aplicación del proceso de aumentado de datos, se tienen en consideración parámetros de variaciones de forma a las imágenes para el aumentado de datos, como rotación, zoom, desplazamiento de anchura y altura, y rotación horizontal. En la segunda etapa, se aplica el modelo de clasificador en cascada y la red neuronal convolucional, empleando el dataset obtenido inicialmente. En la tercera etapa, se evalúa el rendimiento del dataset mediante las técnicas de aprendizaje computacional empleadas, según los aciertos en las detecciones. La metodología se presenta a continuación, en la figura 1.

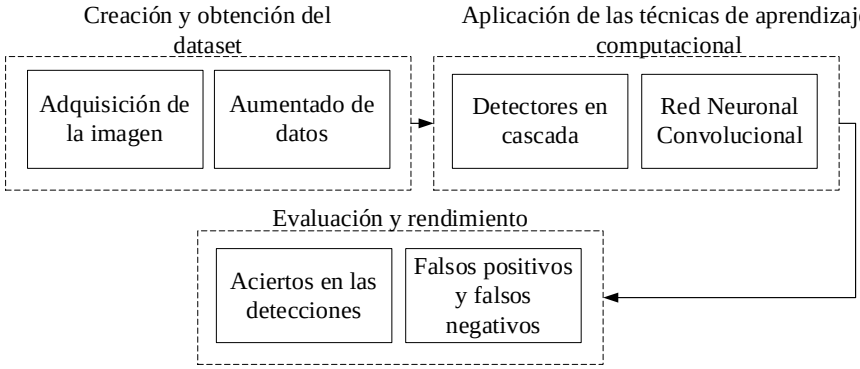

Fig. 1. Metodología propuesta.

2.1. Creación y obtención del dataset

La adquisición de la imagen se realiza en espacios comerciales de la ciudad de Cúcuta, Colombia, mediante una cámara de video de 13 megapíxeles. Las imágenes posteriormente se envían a una computadora personal con procesador Core i7, 4 GB de memoria RAM instalada, sistema operativo de 64 bits con edición Windows 10, donde se realiza el procesamiento. Los elementos de bioseguridad presentes en las imágenes corresponden a tapabocas de tipo higiénico, reutilizables en el caso de los elaborados con tela, así como de un único uso. Además, el dataset también incluye tapabocas de

tipo quirúrgico y de tipo auto-filtrante FFP3. En los tres casos, se incluye una amplia gama de colores, generando así variedad en el conjunto de datos.

Asimismo, en la tabla 1 se presentan los parámetros de caracterización de las zonas en las que se realizó la captura de la imagen. El nivel de luminosidad se obtuvo mediante un luxómetro digital, y estuvo en el rango entre 97 y 4322 luxes, variante entre las 07:00 horas y las 20:00 horas. De igual forma, la altura a la que se ubicó la cámara se encuentra entre 1.7 y 2 metros, obteniendo imágenes en formato JPG y videos en formato MP4.

Tabla 1. Características de la adquisición de la imagen.

Parámetro	Valor
Nivel de luminosidad	[97- 4322] lx
Hora de captura de imagen	[07:00 – 20:00] horas
Altura de localización del dispositivo	[1.70 – 2] m
Formato de imagen/video	JPG/MP4

Para el proceso de aumentado de datos se utilizan los paquetes de Keras y TensorFlow, disponibles para el lenguaje de programación Python [9]. En la tabla 2 se presentan los parámetros requeridos para las modificaciones de forma realizadas sobre las imágenes capturadas inicialmente. Se propone un rango de rotación de hasta 20° para variaciones aleatorias a la imagen. Además, se aplica zoom a la imagen para variar el tamaño aparente de la imagen, en hasta un 17 %. De igual forma, se adaptan variaciones en el desplazamiento de ancho y alto de la imagen con valores de hasta 20 % para cada uno de los casos. Asimismo, se generan imágenes mediante la opción de rotación horizontal, a modo de reflexión en el eje de las X.

Tabla 2. Parámetros de variaciones de forma a las imágenes para el aumentado de datos.

Parámetro	Valor
Angulo de rotación	20°
Zoom	17 %
Desplazamiento de anchura	20 %
Desplazamiento de altura	20 %
Rotación horizontal	Si

2.2. Aplicación de las técnicas de aprendizaje computacional

A continuación, se presentan las estructuras tanto del modelo de clasificador en cascada, como de la red neuronal convolucional.

Clasificador en cascada. Un sistema de clasificadores en cascada se encuentra basado en procesos de aprendizaje automático, donde mediante procesos de entrenamiento, se

evalúa por etapas de modo que se obtenga nueva información sobre los datos de entrada y se genera una función de salida con una clasificación [10]. El modelo de clasificador en cascada se obtiene mediante la aplicación Cascade Trainer GUI, disponible para sistema operativo Windows 7 o superiores [11]. Inicialmente se requieren dos conjuntos de datos correspondientes a imágenes positivas e imágenes negativas. Las imágenes positivas son aquellas en las que se encuentran los tapabocas de tipo higiénico, quirúrgico y auto filtrante, mientras que las imágenes negativas refieren a todos los demás objetos presentes en los ambientes donde se capturó la imagen que no hacen referencia al objeto en cuestión, como vehículos, y prendas de vestir superiores e inferiores; que, para ingresarse a la aplicación, deben alojarse en dos carpetas independientes llamadas "p" para las positivas, y "n" para las negativas.

La aplicación Cascade Trainer GUI en su ventana principal presenta 4 pestañas llamadas Input, Common, Cascade, y Boost, y en cada una de las mismas se realiza la configuración para la obtención del modelo de clasificador. En Input se realiza el cargue de las imágenes positivas y negativas, y se define el porcentaje de imágenes positivas a utilizar para el entrenamiento, que para el presenta caso es del 100 %. Asimismo, en la pestaña Common, se configura entra otros, el número de etapas e hilos, siendo 10 y 5 dichos valores respectivamente. En la pestaña Cascade se configuran los valores de ancho y alto de las muestras, siendo imágenes de tamaño 24x24, además del tipo de caracteristicas, siendo en el presente caso de tipo HAAR, y, en la pestaña Boost, se define el tipo de impulso para el modelo de clasificación, siendo GAB el definido, basado en el Algoritmo Gentle Adaboost, además de la tasa mínima de acierto, definida en 0.995. En la figura 2 se presenta la estructura del modelo de clasificador en cascada, donde se ilustra el cargue de las imágenes a la aplicación Cascade Trainer GUI, y la obtención del archivo .XML que contiene el modelo de clasificación.

Fig. 2. Estructura para la obtención del modelo de clasificador en cascada.

Red neuronal convolucional. Las redes neuronales convolucionales (CNN) son un tipo de red neuronal artificial que se encuentra basada procesos de aprendizaje profundo. El aprendizaje de la red se basa en captar las características únicas de los objetos y generalizarlos [12]. La estructura básica de una CNN consta de 5 etapas denominadas entrada, convolución, activación, muestreo, y obtención de probabilidades [13]. La entrada hace referencia a los pixeles de imagen y características de tamaño y color. En las capas de convolución se procesan las neuronas conectadas mediante pixeles vecinos, y según la cantidad de filtros se determina el volumen de salida. Las funciones de activación refieren a los métodos para propagar la salida de los nodos de una capa hacia las demás capas de la red. En la etapa de muestreo, se realizan las modificaciones de ancho y alto de los datos, sin alterar la profundidad de la

información en proceso. Finalmente, en la etapa de obtención de probabilidades, la red arroja su predicción respecto a la categoría a la que pertenece el objeto presente en la imagen ingresada al sistema [14].

Para el desarrollo de la red neuronal convolucional en lenguaje Python, se requiere de los paquetes de Keras y TensorFlow. En la tabla 3 se presentan los parámetros de configuración de la red neuronal. Se proponen 20 épocas para el entrenamiento de la red, con pasos de 32 imágenes de tamaño 200x200 pixeles. Además, se ejecutan 2 filtros de convolución con tamaños de 2x2 y 3x3 respectivamente. Asimismo, para la reducción en las dimensiones de los datos se utiliza la función MaxPooling, que toma el mayor valor al ejecutarse mediante un filtro de 2x2 [15]. La función de activación utilizada es la de tipo ReLU, y adicionalmente, mediante el Dropout de desactivan instantáneamente el 50 % de las neuronas, evitando así un único camino de entrenamiento. De igual forma, para estimar los valores de las predicciones de salida (persona con tapabocas o persona sin tapabocas) se utiliza la función Softmax.

Tabla 3. Parámetros de configuración de la red neuronal convolucional.

Parámetro	Valor
Épocas	20
Pasos	32
Dimensiones de entrada	200x200
Cantidad de filtros de convolución	2
Tamaño de filtros de convolución	2x2 y 3x3
Tamaño y estructura del Pooling	2x2, MaxPooling
Función de activación	ReLU
Dropout	0.5
Función de probabilidades	Softmax

2.3. Evaluación y rendimiento

Las pruebas al dataset se realizan mediante la medición de falsos positivos y negativos, así como la ponderación de verdaderos positivos y verdaderos negativos en el caso del sistema de clasificador en cascada, y al igual que para la red neuronal convolucional mediante una matriz de confusión se ponderan los de aciertos en las predicciones. Para cada uno de los casos, se calcula el error medio y de esta forma se determina el porcentaje de inconsistencias en las clasificaciones.

3. Resultados

En la tabla 4 se presenta la relación entre el número de imágenes capturadas inicialmente y las obtenidas posterior al aplicar las técnicas de aumentado de datos. Un total de 1,450 imágenes inicialmente se aumentan en aproximadamente 10 veces,

obteniendo 14,067 imágenes, distribuidas en 7,174 imágenes de personas con tapabocas y 6,893 personas sin tapabocas. La eficiencia con la que se realizó el aumentado de las imágenes fue de 89.73 % y 88.81 % respectivamente.

Tabla 4. Relación entre el número de imágenes capturadas y obtenidas posterior al aumentado de datos.

Categoría	Antes	Después	Eficiencia en el aumentado
Con tapabocas	738	7174	89.73%
Sin tapabocas	712	6893	88.81%
Total	1450	14067	

De igual forma, el tiempo empleado para el entrenamiento del sistema de clasificadores en cascada fue de 83 minutos, mientras que la red neuronal convolucional requirió de 247 minutos en entrenarse. El tiempo utilizado en el entrenamiento es dependiente tanto de la estructura de los modelos, como de la herramienta de hardware utilizada en el procesamiento, que, en el presente caso, es una computadora personal con procesador Core i7, 4 GB de memoria RAM instalada y sistema operativo Windows 10.

Así mismo, en la figura 3 se presentan los resultados obtenidos en las predicciones de las técnicas de aprendizaje computacional. Para el modelo de clasificador en cascada, de 73 imágenes de personas sin tapabocas, se logró la estimación correcta en 67 de las mismas, mientras que en 6 oportunidades los clasificó como si portaran el tapabocas. Por otra parte, de 72 personas que portaban el tapabocas, se estimó correctamente la predicción de 59 de ellos, mientras que en 13 oportunidades se categorizó como si las personas no portaran el tapabocas. Asimismo, al aplicar la red neuronal convolucional con el mismo conjunto de imágenes de prueba, se logró la predicción acertada de 72 personas que no portaban el tapabocas, y de 65 personas que si lo portaban.

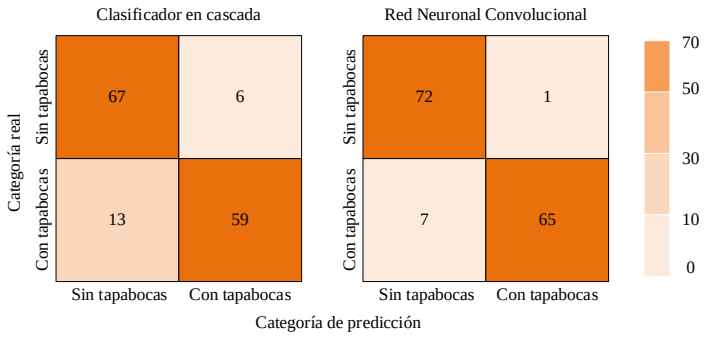

Fig. 3. Matriz de confusión para las técnicas de aprendizaje computacional.

Así mismo, en las ecuaciones 1 y 2 se muestra la forma en cómo se obtuvo el error medio en las predicciones al aplicar el modelo de clasificador en cascada y la red

neuronal convolucional respectivamente. En el sistema de clasificadores en cascada se genera un error de 19.83 %, mientras que, para la red neuronal convolucional, el error fue de 9.81 %:

$$\sqrt{\left(\frac{6}{73}\right)^2 + \left(\frac{13}{72}\right)^2} = 0.1983 \; o \; 19.83\%, \tag{1}$$

$$\sqrt{\left(\frac{1}{73}\right)^2 + \left(\frac{7}{72}\right)^2} = 0.981 \; o \; 9.81\%. \tag{2}$$

4. Conclusiones

Las estructuras expuestas, tanto para el método de aprendizaje automático como para el de aprendizaje profundo, pueden definirse como estructuras comunes en los modelos de aprendizaje computacional, y emulan un comportamiento preliminar del conjunto de datos ante este tipo de entrenamiento, por lo que, mejorando y potenciando las fases de entrenamiento, se potencializa el comportamiento del dataset.

Dicho dataset al ser elaborado con imágenes de contextos y ambientes reales, con niveles de luminosidad variantes a lo largo del día y con tomas desde alturas donde se logra percibir los elementos de bioseguridad facial utilizados por las personas, y probados en espacios comerciales de la ciudad de Cúcuta, presenta una robustez en el comportamiento de las técnicas de aprendizaje computacional utilizadas, ya que logran aciertos en la clasificación superiores al 80 %.

Asimismo, la red neuronal convolucional mejora en aproximadamente un 10 % el comportamiento en las clasificaciones respecto al modelo de clasificador en cascada, generado por el número de etapas de entrenamiento y procesos computacionales propios de los procesos de aprendizaje profundo.

Referencias

1. Shereen, M.A., Khan, S., Kazmi, A., Bashir, N., Siddique, R.: COVID-19 infection: Origin, transmission, and characteristics of human coronaviruses (2020). https://doi.org/10.1016/j.jare.2020.03.005.
2. Zou, X.: A Review of object detection techniques. In: Proc. Int. Conf. Smart Grid Electr. Autom. ICSGEA, pp. 251–254 (2019). https://doi.org/10.1109/ICSGEA.2019.00065.
3. Ramírez Jiménez, D., Quintero Ospina, J.D.: Clasificación de patologías presentes en la columna vertebral mediante técnicas de máquinas de aprendizaje. Ing. Solidar. 15, 1–23 (2019). https://doi.org/10.16925/2357-6014.2019.01.05.
4. Selvaraj, K., Fathima, A.A., Vaidehi, V.: Multi-class object detection by part based approach. In: International Conference on Recent Trends in Information Technology, ICRTIT 2012. pp. 114–118 (2012). https://doi.org/10.1109/ICRTIT.2012.6206837.
5. Niño Rondón, C.V., Castro Casadiego, S.A., Medina Delgado, B., Guevara Ibarra, D., Ramirez Mateus, J.J., Puerto López, K.C.: Comparación multiplaforma de técnicas basadas en visión artificial para detección de personas en espacios abiertos. Investig. e Innovación en Ing. 9, 22–33 (2021). https://doi.org/10.17081/invinno.9.1.3965.

6. Shorten, C., Khoshgoftaar, T.M.: A survey on Image Data Augmentation for Deep Learning. J. Big Data. 6 (2019). https://doi.org/10.1186/s40537-019-0197-0.
7. Plebani, E., Celona, L., Pau, D., Karimi, P., Marcon, M.: Training an object detector using only positive samples. In: 2015 IEEE 1st International Workshop on Consumer Electronics - Novi Sad, CE WS 2015. pp. 1–4. Institute of Electrical and Electronics Engineers Inc. (2017). https://doi.org/10.1109/CEWS.2015.7867139.
8. Mikołajczyk, A., Grochowski, M.: Data augmentation for improving deep learning in image classification problem. In: 2018 International Interdisciplinary PhD Workshop, IIPhDW 2018. pp. 117–122. Institute of Electrical and Electronics Engineers Inc. (2018). https://doi.org/10.1109/IIPHDW.2018.8388338.
9. Kakuda, K., Enomoto, T., Miura, S.: Nonlinear activation functions in CNN based on fluid dynamics and its applications. C. - Comput. Model. Eng. Sci. 118, 1–14 (2019). https://doi.org/10.31614/cmes.2019.04676.
10. Siqueira, D.L., Manso Correa Machado, A.: People Detection and Tracking in Low Frame-rate Dynamic Scenes. IEEE Lat. Am. Trans. 14, 1966–1971 (2016). https://doi.org/10.1109/TLA.2016.7483541.
11. Phase, T.R., S. Patil, S.: Building Custom HAAR-Cascade Classifier for face Detection. Int. J. Eng. Res. V. 8 (2020). https://doi.org/10.17577/ijertv8is120350.
12. Öztürk, Ş., Akdemir, B.: Effects of Histopathological Image Pre-processing on Convolutional Neural Networks. In: Procedia Computer Science. pp. 396–403. Elsevier B.V. (2018). https://doi.org/10.1016/j.procs.2018.05.166.
13. Krishna Sai, B.N., Sasikala, T.: Object Detection and Count of Objects in Image using Tensor Flow Object Detection API. In: Proceedings of the 2nd International Conference on Smart Systems and Inventive Technology, ICSSIT 2019. pp. 542–546. Institute of Electrical and Electronics Engineers Inc. (2019). https://doi.org/10.1109/ICSSIT46314.2019.8987942.
14. Agarap, A.F.: Deep Learning using Rectified Linear Units (ReLU). arXiv. (2018).
15. Suárez Paniagua, V., Segura Bedmar, I.: Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. BMC Bioinformatics. 19, 209 (2018). https://doi.org/10.1186/s12859-018-2195-1.