

Universidad Politécnica del Estado de Morelos

VERITAS PROYECTO INTEGRADOR SISTEMAS INTELIGENTES

**CEDILLO MEJIA ANGEL ALEJANDRO
GUERRERO ARZATE VICTOR SAHID
SOTELO RODRIGUEZ KEVIN**

JUAN PAULO SÁNCHEZ HERNÁNDEZ

ÍNDICE

01. Introducción
02. Objetivo
03. Diseño de la propuesta
04. Implementación
05. Demostración
06. Conclusiones
07. Referencias bibliográficas

INTRODUCCIÓN

Bajo la premisa de poder ayudar a las personas principalmente a la detección de contenido falso o manipulado.

Una herramienta práctica que ayude a las personas a discernir entre lo real y lo generado por un código

OBJETIVO

Desarrollar un sistema inteligente basado en redes neuronales convolucionales (CNN) que permita clasificar imágenes como reales o generadas por inteligencia artificial, para apoyar la detección automatizada de contenido sintético y contribuir a la protección de la integridad informativa en entornos digitales.



DISEÑO DE LA PROPUESTA



Nuestra interfaz es minimalista cumpliendo con su función principal de seleccionar una imagen para posteriormente clasificarla

Teniendo un botón extra para poder tomar tu una foto directamente.

IMPLEMENTACIÓN

```
# Definir los hiperparametros de la arquitectura CNN
epocas = 20 # Numero de epocas
altura, anchura = 512, 512 # Dimensiones de las imagenes
batch_size = 20 # Tamaño del batch
pasos = 100 # Pasos por epoca
```

ÉPOCAS (EPOCAS = 20):

GARANTIZA QUE EL MODELO TENGA SUFICIENTES ITERACIONES PARA APRENDER PATRONES.

DIMENSIONES DE LAS IMÁGENES (ALTURA, ANCHURA = 512, 512):

FACILITA EL PROCESAMIENTO Y REDUCE LA COMPLEJIDAD COMPUTACIONAL.

TAMAÑO DEL BATCH (BATCH_SIZE = 20):

MEJORA LA EFICIENCIA DEL ENTRENAMIENTO AL PROCESAR DATOS EN BLOQUES.

PASOS POR ÉPOCA (PASOS = 100):

CONTROLA LA CANTIDAD DE DATOS UTILIZADOS EN CADA ITERACIÓN COMPLETA DEL CONJUNTO DE ENTRENAMIENTO.

IMPLEMENTACIÓN

```
entrenar = ImageDataGenerator(  
    rescale=1/255,  
    zoom_range=0.3,  
    horizontal_flip=True,  
    rotation_range=30, # Rotar las imágenes  
    width_shift_range=0.2, # Desplazar horizontalmente  
    height_shift_range=0.2, # Desplazar verticalmente  
    brightness_range=[0.8, 1.2] # Ajustar el brillo  
)
```

TÉCNICAS QUE MEJORAN LA CAPACIDAD DEL MODELO PARA GENERALIZAR, REDUCIENDO EL RIESGO DE SOBREAJUSTE Y AUMENTANDO LA ROBUSTEZ FRENTE A DATOS NO VISTOS.

IMPLEMENTACIÓN

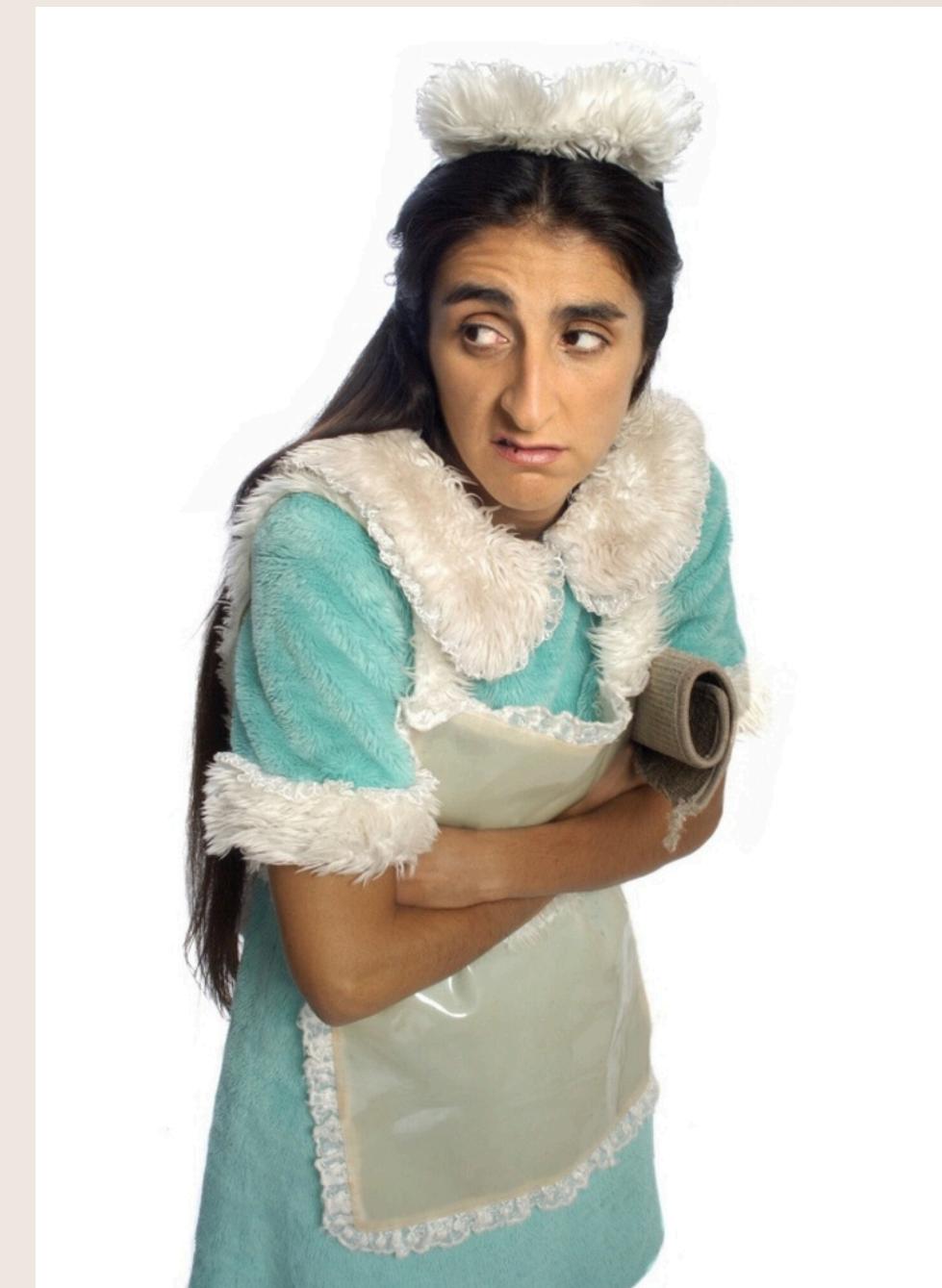
```
# Sección 6 de código
# Entrenar el modelo
# Callback para detener el entrenamiento si no hay mejoras
early_stopping = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)

# Callback para guardar el mejor modelo
model_checkpoint = ModelCheckpoint('Modelo/mejor_modelo.keras', save_best_only=True, monitor='val_loss')

# Entrenar el modelo con callbacks
ModeloCNN.fit(
    imagenes_entrenamiento,
    validation_data=imagenes_validacion,
    epochs=epocas,
    validation_steps=pasos,
    verbose=1,
    callbacks=[early_stopping, model_checkpoint]
)
```

UTILIZANDO TÉCNICAS PARA OPTIMIZAR EL RENDIMIENTO Y EVITAR PROBLEMAS COMUNES COMO EL SOBREAJUSTE. SE EMPLEARON CALLBACKS PARA MEJORAR LA EFICIENCIA DEL ENTRENAMIENTO Y GARANTIZAR LA CALIDAD DEL MODELO FINAL.

DEMOSTRACIÓN



CONCLUSIONES

ESTE PROYECTO DESARROLLÓ UN CLASIFICADOR DE
IMÁGENES QUE DISTINGUE ENTRE IMÁGENES
GENERADAS POR IA Y REALES.

UTILIZA UNA ARQUITECTURA CNN OPTIMIZADA,
GARANTIZANDO UN MODELO EFICIENTE Y PRECISO.

REFERENCIAS

CHOLLET, F. (2018). DEEP LEARNING WITH PYTHON (2ND ED.). MANNING PUBLICATIONS.

GÉRON, A. (2019). HANDS-ON MACHINE LEARNING WITH SCIKIT-LEARN, KERAS, AND TENSORFLOW: CONCEPTS, TOOLS, AND TECHNIQUES TO BUILD INTELLIGENT SYSTEMS (2ND ED.). O'REILLY MEDIA.

RAWAT, W., & WANG, Z. (2017). DEEP CONVOLUTIONAL NEURAL NETWORKS FOR IMAGE CLASSIFICATION: A COMPREHENSIVE REVIEW. NEURAL COMPUTATION, 29(9), 2352-2449. [HTTPS://DOI.ORG/10.1162/NECO_A_00990](https://doi.org/10.1162/NECO_A_00990)

ABADI, M., BARHAM, P., CHEN, J., CHEN, Z., DAVIS, A., DEAN, J., ... & ZHENG, X. (2016). TENSORFLOW: A SYSTEM FOR LARGE-SCALE MACHINE LEARNING. PROCEEDINGS OF THE 12TH USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION (OSDI 16), 265-283. [HTTPS://WWW.USENIX.ORG/SYSTEM/FILES/CONFERENCE/OSDI16/OSDI16-ABADI.PDF](https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf)

MEHDIPOUR GHAZI, M., KHALIGH-RAZAVI, S. M., & NABI, M. (2018). DIAGNOSING SKIN LESIONS USING DEEP CONVOLUTIONAL NEURAL NETWORKS. PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON MEDICAL IMAGE COMPUTING AND COMPUTER-ASSISTED INTERVENTION (MICCAI), 85-93. [HTTPS://DOI.ORG/10.1007/978-3-030-00928-1_10](https://doi.org/10.1007/978-3-030-00928-1_10)