

Universidad de Oriente.

Núcleo de Anzoátegui.

Departamento de Ingeniería en Computación y Sistemas .

Taller de Computación Gráfica

Sección: 01.



## **Caricatura. Avance Final**

Profesor:

Pedro Dorta.

Estudiante:

Angel Osuna. / CI: 28.407.265.

Diego Alvarez. / C.I: 30.983.861.

Javier Mujica. / C.I: 30.711.587.

Mario Rupprecht. / C.I: 30.433.106.

Barcelona, 26 de julio de 2024.

## Informe sobre el Proyecto de Caricaturización de Imágenes

Este proyecto es una aplicación de caricaturización de imágenes que utiliza técnicas de procesamiento de imágenes para transformar fotos de personas en caricaturas. Implementa funcionalidades para redimensionar características faciales, como los ojos, y aplica efectos de caricatura a las imágenes usando OpenCV y herramientas de interfaz gráfica como Tkinter y PIL.

### Funciones Clave

#### 1. Redimensionar Ojos (`redimensionar_ojos`):

- **Descripción:** Esta función detecta los ojos en una imagen y los redimensiona para hacerlos más grandes, dando un efecto caricaturizado.
- **Proceso:**
  - Carga la imagen y la convierte a escala de grises.
  - Utiliza un clasificador en cascada para detectar ojos.
  - Calcula el tamaño promedio de los ojos detectados y amplía cada ojo a ese tamaño.
  - Crea una máscara elíptica para suavizar los bordes de los ojos redimensionados y combina estos con la imagen original.
- **Impacto en la Caricatura:** Los ojos ampliados contribuyen a un aspecto exagerado y caricaturesco, acentuando una de las características faciales más destacadas.

## 2. Caricaturización de Ojos (ojos\_caricatura):

- **Descripción:** Similar a la función para redimensionar ojos, pero también aplica un filtro de k-means para suavizar los colores.
- **Proceso:**
  - Detecta y redimensiona los ojos como en la función anterior.
  - Aplica k-means para suavizar los colores del área de los ojos.
- **Impacto en la Caricatura:** Acentúa aún más el efecto de caricatura en los ojos mediante la suavización del color.

## 3. Conversión a Caricatura (convertir\_a\_caricatura):

- **Descripción:** Transforma la imagen original en una caricatura aplicando una serie de filtros y técnicas.
- **Proceso:**
  - Aplica un desenfoque mediano a la imagen en escala de grises.
  - Usa umbralización adaptativa para resaltar los bordes.
  - Realiza una segmentación de color para obtener la piel y suaviza los colores usando k-means.
- **Impacto en la Caricatura:** Define los contornos y los colores básicos para lograr un efecto de caricatura general.

## 4. Detección y Dibujo de Contornos (detectar\_y\_dibujar\_contornos):

- **Descripción:** Detecta los bordes en la imagen para mejorar la definición de los contornos.
- **Proceso:**
  - Aplica un desenfoque gaussiano para suavizar la imagen.
  - Usa umbralización adaptativa para detectar los bordes.

- **Impacto en la Caricatura:** Mejora la nitidez y definición de los bordes en la caricatura.

#### 5. Proceso Final (`proceso_final`):

- **Descripción:** Aplica una serie de pasos finales para refinar la imagen caricaturizada.
- **Proceso:**
  - Llama a la función de detección y dibujo de contornos.
  - Encuentra contornos significativos y los dibuja sobre la imagen.
- **Impacto en la Caricatura:** Asegura que la caricatura final tenga bordes bien definidos y un aspecto general mejorado.

#### 6. Procesamiento de Imagen (`procesar_imagen` y `procesar_imagen_experimental`):

- **Descripción:** Permite al usuario seleccionar una imagen, aplicar los efectos de caricatura y guardar el resultado.
- **Proceso:**
  - Abre un cuadro de diálogo para seleccionar y guardar la imagen.
  - Aplica las funciones de caricaturización y muestra la imagen final.
- **Impacto en la Caricatura:** Facilita la interacción del usuario con el sistema y la obtención de imágenes caricaturizadas.

#### 7. Interfaz de Usuario:

- **Descripción:** Usa Tkinter para crear una interfaz gráfica que permita al usuario iniciar el proceso de caricaturización.
- **Componentes:**

- Botones para activar el modo experimental y el modo estándar de caricaturización.
- Mensajes de información para guiar al usuario sobre el uso del software.
- **Impacto en la Caricatura:** Proporciona una experiencia de usuario intuitiva y accesible.

## Instrucciones para Ejecutar el Código

Para ejecutar el código de caricaturización de imágenes descrito, seguir estos pasos:

### 1. Preparativos Iniciales

**Instalación de Dependencias:** Asegúrate de tener instaladas las bibliotecas necesarias. Puedes instalar las dependencias usando pip. Abre una terminal o línea de comandos y ejecuta:

```
pip install opencv-python-headless numpy pillow
```

- **Archivos Necesarios:**

- **Imagen de Fondo:** Asegúrate de tener una imagen de fondo para la interfaz gráfica, como Proyecto .png. Coloca esta imagen en el mismo directorio donde se encuentra el archivo de código.

- **Clasificadores de Haar:** OpenCV incluye los clasificadores de Haar necesarios (`haarcascade_eye.xml` y `haarcascade_mcs_mouth.xml`). Estos están generalmente incluidos en la instalación de OpenCV, pero si no están disponibles, asegúrate de tenerlos en la ubicación correcta (`cv2.data.haarcascades`).

## 2. Estructura del Código

- El código está dividido en varias funciones para manejar diferentes aspectos del proceso de caricaturización, como redimensionar ojos, convertir la imagen a estilo caricatura, y mostrar la interfaz gráfica.

## 3. Ejecución del Código

- Guarda el código en un archivo con extensión `.py`, por ejemplo, `caricatura.py`.
- Abre una terminal o línea de comandos.
- Navega al directorio donde guardaste el archivo `.py`.

Ejecuta el archivo Python con el siguiente comando:

```
python caricatura.py
```

## 4. Uso de la Aplicación

- **Interfaz Gráfica:**

- Al ejecutar el código, se abrirá una ventana con la interfaz gráfica creada con Tkinter.
- La interfaz tendrá dos botones:
  - **"Experimental"**: Activa el modo experimental para aplicar caricaturización con efectos adicionales.
  - **"Crear Caricatura"**: Activa el modo estándar de caricaturización.
- **Selección de Imagen:**
  - Al hacer clic en uno de los botones, se abrirá un cuadro de diálogo para seleccionar una imagen desde tu sistema.
  - Elige una imagen de una persona que desees caricaturizar.
- **Aplicación de Efectos:**
  - El código procesará la imagen seleccionada aplicando los efectos de caricatura.
  - Después de aplicar los efectos, se abrirá una ventana con la imagen caricaturizada.
- **Guardar Imagen:**
  - Tras ver la imagen caricaturizada, se abrirá un cuadro de diálogo para guardar la imagen en tu sistema.
  - Elige la ubicación y el nombre del archivo para guardar la imagen resultante.

## 5. Notas Adicionales

- **Ajustes:** Puedes ajustar los parámetros en el código para afinar los efectos de caricatura según tus necesidades.

- **Resolución de Imágenes:** Para obtener los mejores resultados, usa imágenes con una resolución adecuada y asegúrate de que el rostro esté claramente visible en la imagen seleccionada.
- **Compatibilidad:** Este código está diseñado para ejecutarse en entornos donde estén instaladas las bibliotecas OpenCV y Tkinter. Asegúrate de que tu entorno de Python esté configurado correctamente.

## Conclusiones

El proyecto combina técnicas de procesamiento de imágenes avanzadas con una interfaz de usuario amigable para crear caricaturas de forma automática. Las funciones clave, como la redimensión de ojos, y la aplicación de filtros de caricatura, trabajan en conjunto para transformar las imágenes en versiones estilizadas y exageradas, características de las caricaturas. La interfaz gráfica facilita el uso del programa, haciendo accesible la caricaturización para usuarios sin experiencia técnica

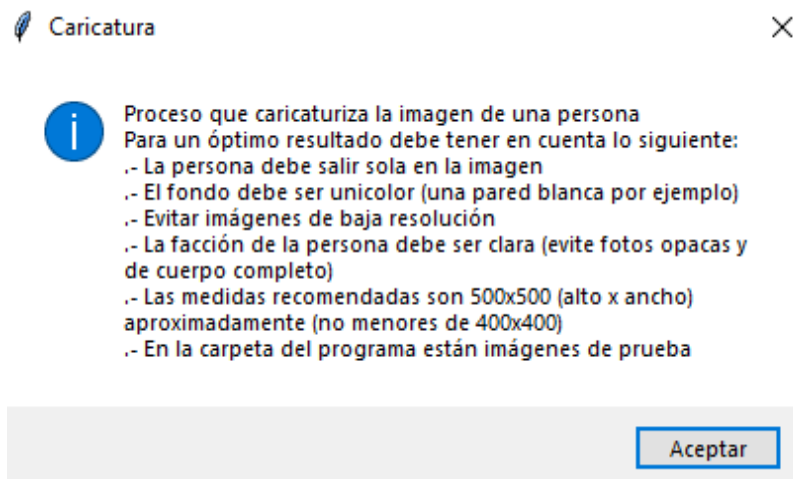


## Interfaz Grafica

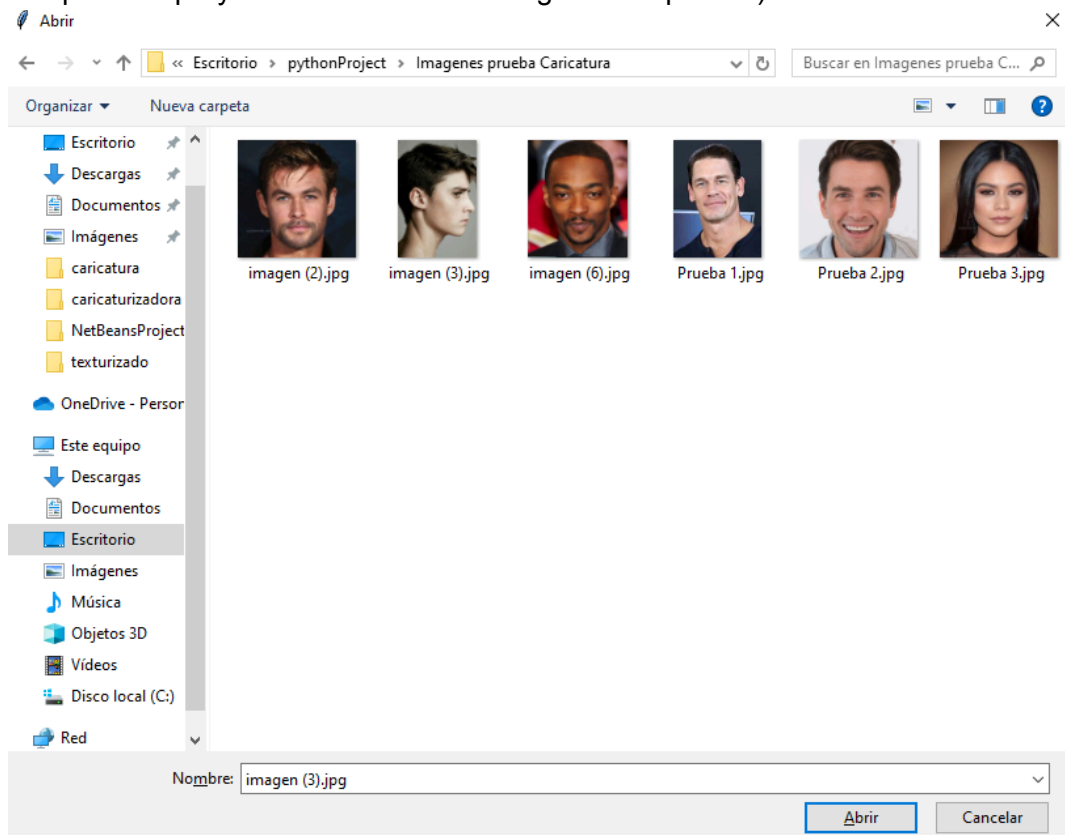


### Ventana principal que contiene dos botones

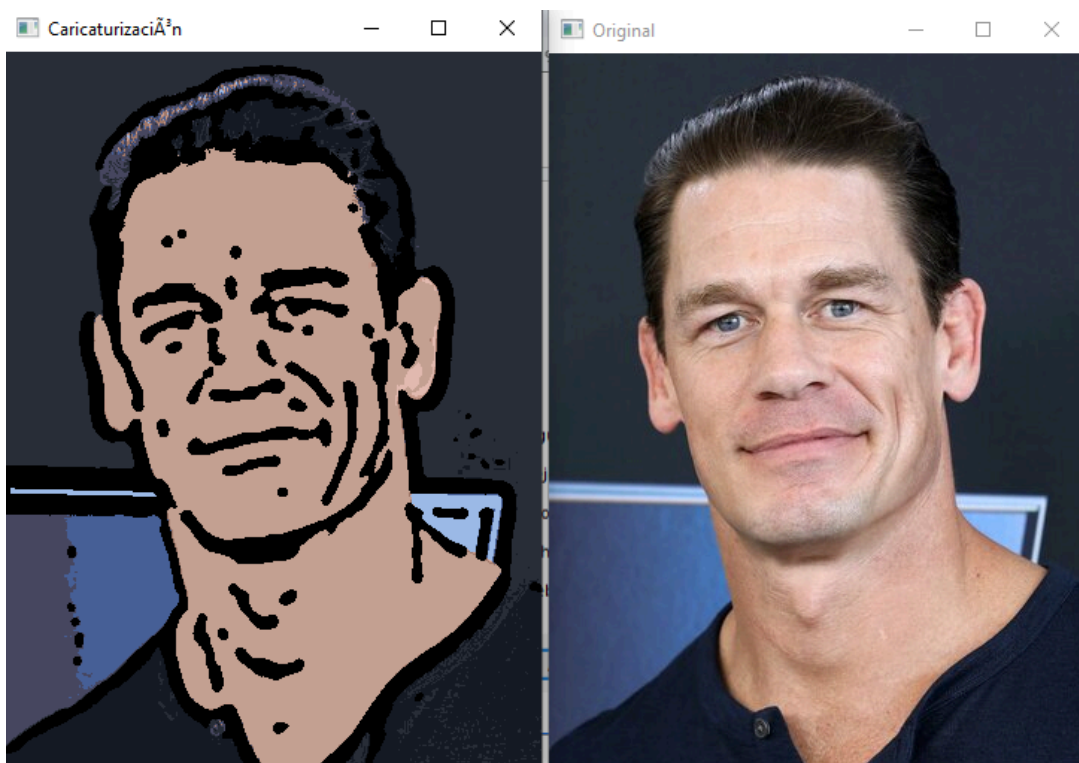
1.- Crear Caricatura: Al darle click salta un cuadro de diálogo indicando las pautas para conseguir el mejor resultado



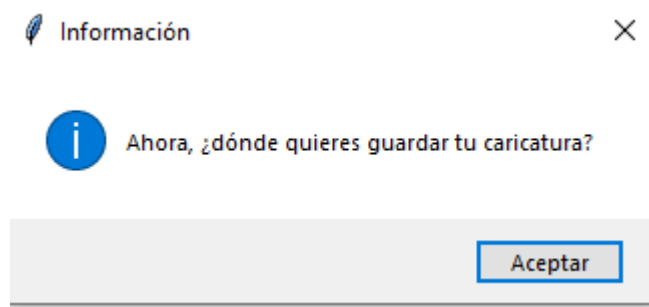
Al darle aceptar abre una carpeta para elegir la imagen que se desea caracterizar  
(en la carpeta del proyecto se encuentran imágenes de prueba)



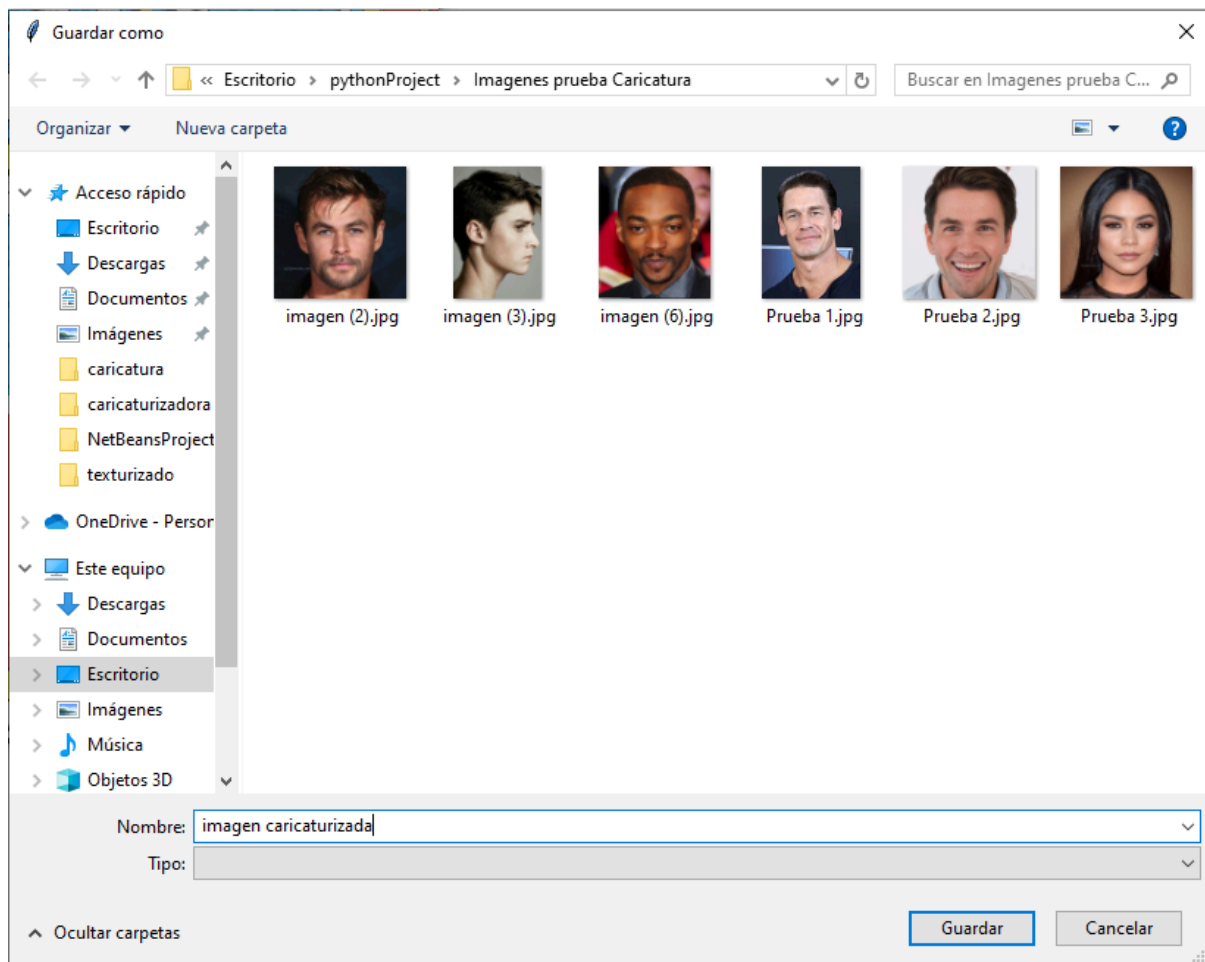
Una vez seleccionada la imagen se le aplicarán los procesos que se mencionaron anteriormente para conseguir el efecto de caricatura en la imagen, para posteriormente se mostrarán ambas imágenes, como era originalmente y como se ve ahora que se le aplicaron los procesos.



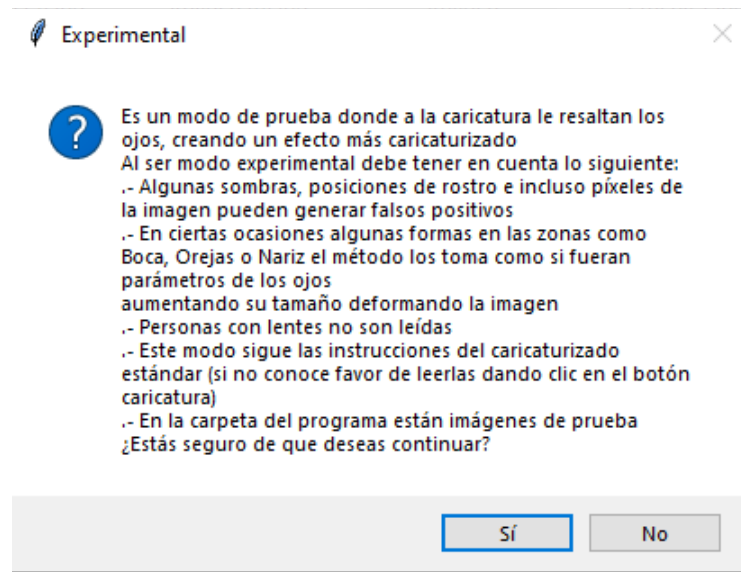
Luego se le indica al usuario donde desea guardar la imagen.



Ahora el usuario puede indicar el lugar y donde este quiere guardar su imagen modificada y el nombre que tendrá esta.



2.-Experimental: En tal caso de presionar el botón experimental aparecerá este mensaje de advertencia.



Al darle al botón si, todo lo anterior se hará de la misma manera, con la diferencia que ahora se aplicó la redimensión a los ojos. Obteniendo este resultado:

