

# Búsqueda y reemplazo de objetos

Integrantes:

Almanza Contreras Jose de Jesus  
Alonso Romero Pablo Emilio  
Avila Monjaraz Victor Hassiel  
Durán Torres Leonardo Gael



# Descripción del problema

## ¿Qué problema resolvemos?

El proyecto tiene como objetivo detectar un objeto específico dentro de una imagen principal y reemplazarlo por otro objeto proporcionado por el usuario.



Cargar imágenes → Buscar objeto → Verificar → Reemplazar

## ¿Cómo se lleva a cabo?

- Se utilizan tres imágenes:
  - Imagen principal (fondo).
  - Imagen del objeto a buscar.
  - Imagen del objeto a insertar.
- El algoritmo busca la posición del objeto dentro de la imagen principal comparando píxel por píxel.
- Si se encuentra el objeto:
  - Se reemplaza por la nueva imagen.
  - Si las imágenes no son del mismo tamaño, se redimensiona la imagen a insertar.
- Si no se encuentra:
  - Se imprime o retorna un mensaje de error indicando que no se detectó el objeto.

## Metodología paso a paso



# Metodología paso a paso

## Proceso general del sistema

# Ejemplo 1



Imagen de fondo - Galeria 1



Imagen a reemplazar - Galería 1

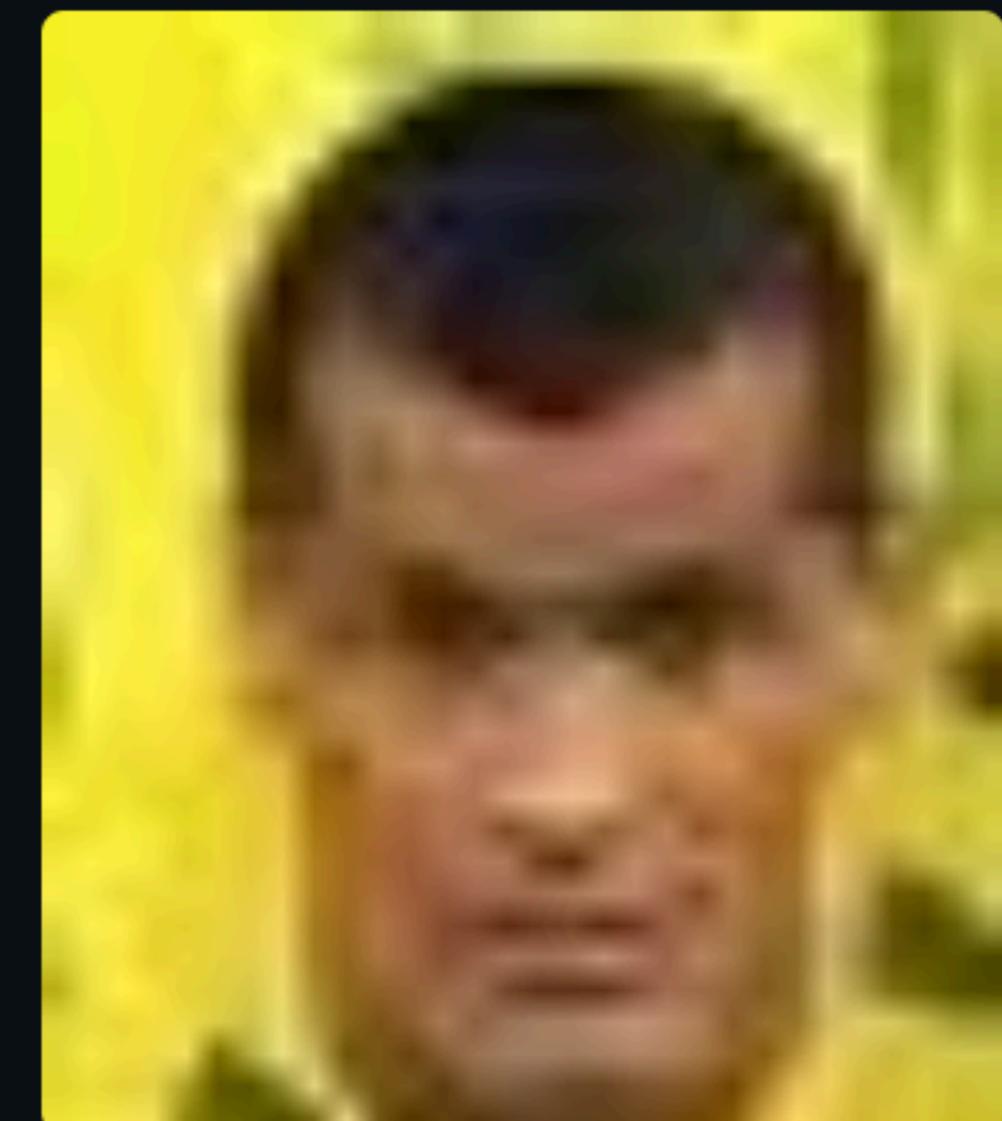


Imagen a buscar - Galería 1



Imagen con reemplazo - Galería 1

# Ejemplo 2



Imagen de fondo - Galeria 3



Imagen a buscar - Galeria 3



Imagen a reemplazar - Galeria 3



Imagen con reemplazo - Galeria 3

# Ejemplo 3



Imagen de fondo - Galería 4



Imagen a reemplazar - Galería 4



Imagen a buscar - Galería 4

No se encontró el pedazo de imagen a buscar  
en la imagen de fondo.

# Resultados Obtenidos

11 —



*Casos donde funcionó correctamente:*

- *Cuando las imágenes están en formato .png y en modo RGB.*
- *Al hacer recortes desde Paint, se detectan con precisión.*
- *Redimensionamiento del objeto funciona bien si el tamaño original no es muy distinto.*

# Resultados Obtenidos

11 —



*Problemas encontrados:*

- *Formatos incompatibles: Imágenes con formato distinto a .png pueden fallar.*
- *Herramientas de recorte: Usar la herramienta de recortes de Windows genera valores en decimales (0–1) en lugar de 0–255.*
- *Conversión incorrecta: Convertidores en línea no preservan bien los datos, lo que causa errores en la detección.*

# Resultados Obtenidos

11



## 💡 Solución aplicada:

- Conversión manual a .png y recortes realizados exclusivamente con Paint para mantener fidelidad en los valores RGB.



# Conclusión

El proyecto permitió identificar y reemplazar objetos en imágenes comparando píxeles.

Funcionó bien con imágenes PNG en formato RGB, pero presentó fallos con otros formatos.

Se aprendió la importancia del formato y la fidelidad de los datos en el procesamiento de imágenes.

An aerial photograph showing a dense forest of coniferous trees covering a hillside. The forest is thick and green, with some lighter-colored areas where sunlight might be filtering through or where different types of vegetation are growing. The terrain appears slightly hilly and uneven.

Gracias por su  
atención

