

Reporte de Práctica: Detección de Color y Filtrado en Tiempo Real

Nombre: Eduardo Correa Flores

Materia: Visión Artificial

Tema: Segmentación por color y técnicas de suavizado

Lenguaje: Python

Objetivo:

Implementar un sistema de captura en tiempo real que:

1. Detecte objetos de color rojo mediante segmentación HSV.
 2. Compare diferentes técnicas de suavizado para eliminar ruido.
 3. Analice la eficacia de cada filtro en la calidad de la imagen resultante.
-

Descripción de la Práctica:

1. Configuración Inicial

python

Copy

Download

```
cap = cv2.VideoCapture(0)
```

- Se inicializa la cámara web (dispositivo 0).

2. Procesamiento por Fotogramas

python

Copy

Download

```
while(1):
```

```
    _, frame = cap.read()
```

```
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
```

- Cada fotograma se convierte de **BGR a HSV** para facilitar la detección de color.

3. Segmentación del Color Rojo

python

Copy

Download

```
lower_red = np.array([100, 150, 30])
```

```
upper_red = np.array([255, 255, 150])
```

```
mask = cv2.inRange(hsv, lower_red, upper_red)
```

```
res = cv2.bitwise_and(frame, frame, mask=mask)
```

- **Máscara HSV:** Define el rango de color rojo (valores ajustables).
- `bitwise_and`: Aplica la máscara para aislar el color en el fotograma original.

4. Técnicas de Suavizado

Se comparan cuatro métodos:

Filtro	Código	Propósito
Filtro Promedio	<code>cv2.filter2D(res, -1, kernel)</code>	Suavizado uniforme con kernel 15x15.
Gaussiano	<code>cv2.GaussianBlur(res, (15,15), 0)</code>	Elimina ruido preservando bordes.
Mediana	<code>cv2.medianBlur(res, 15)</code>	Efectivo contra ruido "sal y pimienta".
Bilateral	<code>cv2.bilateralFilter(res, 15, 75, 75)</code>	Preserva bordes mientras suaviza.

5. Visualización

python

Copy

Download

```
cv2.imshow('frame', frame)    # Original
```

```
cv2.imshow('res', res)        # Segmentación roja
```

```
cv2.imshow('Gaussian Blurring', blur)
```

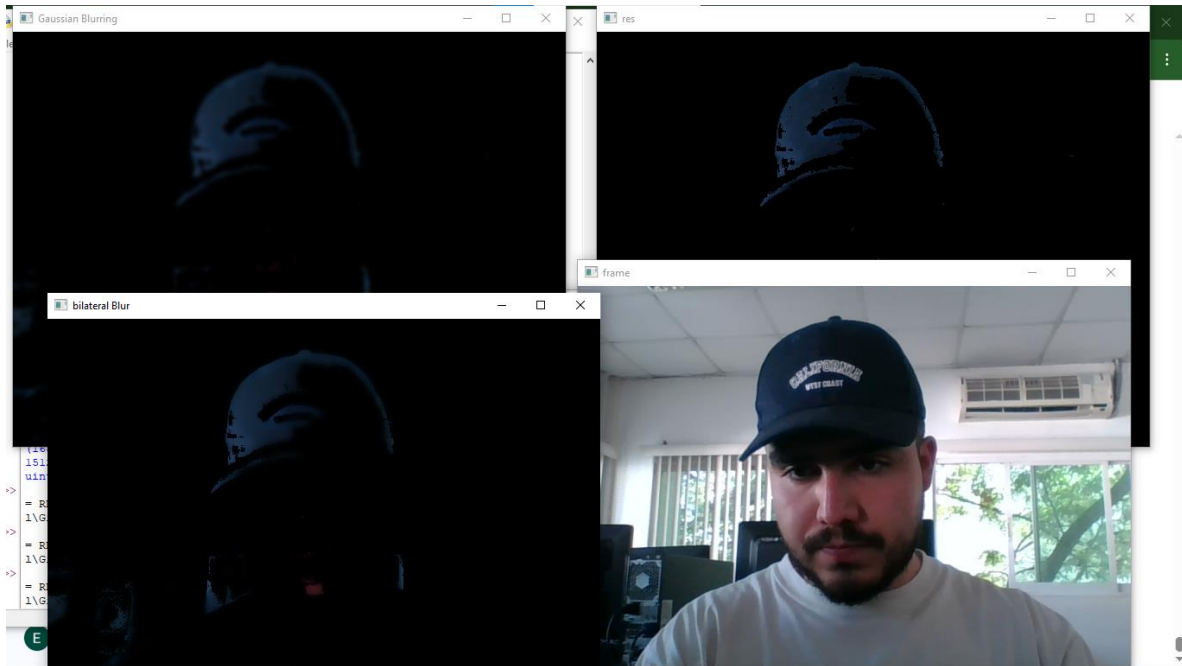
```
cv2.imshow('Median Blur', median)
```

```
cv2.imshow('bilateral Blur', bilateral)
```

- Se muestran los resultados en ventanas separadas.
- **Tecla** ESC (27) para cerrar el programa.

Resultados Obtenidos

Salida	Descripción
frame	Fotograma original de la cámara.
res	Objetos rojos aislados (máscara aplicada).
Gaussian Blurring	Suavizado con distribución gaussiana.
Median Blur	Eliminación de ruido tipo "sal y pimienta".
bilateral Blur	Suavizado que preserva bordes nítidos.



Conclusión

- **Segmentación HSV:** Efectiva para detectar colores específicos en tiempo real.
- **Filtro Gaussiano:** Ideal para suavizado general sin alto costo computacional.
- **Filtro Bilateral:** Mejor opción cuando se requiere preservar detalles de bordes.
- **Aplicaciones:** Robótica (seguimiento de objetos), sistemas de seguridad (detección de intrusos), etc.