

## 1. Importación de librerías

python

Copiar código

```
import cv2
import mediapipe as mp
import pygame
```

### ¿Qué hace?

- **cv2**: Para capturar y mostrar video (OpenCV).
  - **mediapipe**: Para detectar y rastrear manos en tiempo real.
  - **pygame**: Para reproducir sonidos cuando se detectan movimientos específicos.
- 

## 2. Inicialización de MediaPipe Hands

python

Copiar código

```
mp_hands = mp.solutions.hands
mp_drawing = mp.solutions.drawing_utils
```

### ¿Qué hace?

- **mp\_hands**: Accede al módulo de detección de manos.
  - **mp\_drawing**: Permite dibujar las conexiones y puntos clave de la mano en el video.
- 

## 3. Inicialización de sonidos

python

Copiar código

```
pygame.mixer.init()

sounds = [
    pygame.mixer.Sound("#fa.wav"),      # Índice izquierdo
```

```

pygame.mixer.Sound("la.wav"),      # Medio izquierdo
pygame.mixer.Sound("re.wav"),      # Anular izquierdo
pygame.mixer.Sound("#do.wav"),     # Índice derecho
pygame.mixer.Sound("#sol.wav"),    # Medio derecho
pygame.mixer.Sound("si.wav"),      # Anular derecho
]

```

### 🎵 ¿Qué hace?

- Inicia el sistema de audio de pygame.
  - Carga 6 sonidos (3 para cada mano) asociados a distintos dedos.
- 

## 👉 4. Función para detectar si un dedo está abajo

python

Copiar código

```

def is_finger_down(landmarks, finger_tip, finger_mcp):
    return landmarks[finger_tip].y > landmarks[finger_mcp].y

```

### 🧠 ¿Qué hace?

- Compara la posición del dedo **en vertical (eje Y)**:
    - Si la punta del dedo (**tip**) está **más abajo** que su base (**mcp**), el dedo está "doblado".
    - Esto se usa para detectar si el dedo se baja → se toca una nota.
- 

## 📷 5. Captura de video

python

Copiar código

```

cap = cv2.VideoCapture(0)

```

### 📷 ¿Qué hace?

- Inicia la cámara para capturar el video en tiempo real.
- 

## 6. Lógica principal del programa (loop)

python

Copiar código

```
with mp_hands.Hands(...) as hands:
    finger_states = [False] * 6
    while cap.isOpened():
```

### ¿Qué hace?

- Usa un modelo de MediaPipe para detectar hasta 2 manos.
  - Inicializa `finger_states`: guarda el estado (levantado o no) de cada uno de los 6 dedos usados.
  - Entra en un bucle de procesamiento en tiempo real.
- 



## 7. Procesamiento del video y detección de manos

python

Copiar código

```
ret, frame = cap.read()
frame = cv2.flip(frame, 1)
rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
results = hands.process(rgb_frame)
```

### ¿Qué hace?

- Lee un cuadro del video.
- Lo **invierte horizontalmente** (efecto espejo).
- Lo convierte a **RGB** (MediaPipe lo requiere).
- Pasa la imagen al modelo de MediaPipe para detectar las manos.

---

## 👋 8. Detección de dedos activos y reproducción de sonido

```
for h, hand_landmarks in enumerate(results.multi_hand_landmarks):
    mp_drawing.draw_landmarks(...)
    ...
    for i in range(3):
        finger_index = i + h*3
        if is_finger_down(...):
            if not finger_states[finger_index]:
                sounds[finger_index].play()
                finger_states[finger_index] = True
            else:
                finger_states[finger_index] = False
```

### 🧠 ¿Qué hace?

- Itera sobre cada mano detectada ( $h = 0$  izquierda,  $h = 1$  derecha).
- Dibuja los puntos de la mano.
- Itera por **3 dedos** por mano:
  - Dedos índice (8), medio (12), anular (16)
  - MCPs (base del dedo): 5, 9, 13
- Calcula `finger_index` para mapear el dedo con su sonido correspondiente.
- Si el dedo está abajo y **no estaba antes** → reproduce el sonido.

**finger\_index fórmula:**

$i + h*3 \rightarrow$

Mano izquierda ( $h = 0$ ): 0,1,2

Mano derecha ( $h = 1$ ): 3,4,5

---

## 🖼️ 9. Mostrar imagen y salir con 'q'

```
cv2.imshow('Hand Tracking', frame)
```

```
if cv2.waitKey(1) & 0xFF == ord('q'):  
    break
```

### 👁️ ¿Qué hace?

- Muestra el video en una ventana.
  - Si se presiona la tecla **q**, termina el programa.
- 

## 10. Liberar recursos

```
cap.release()  
cv2.destroyAllWindows()
```

### 🧹 ¿Qué hace?

- Libera la cámara.
  - Cierra todas las ventanas de OpenCV.
- 

## Resumen visual de flujo general

```
[ Video (webcam) ]  
    ↓  
[ Flip + RGB Conversion ]  
    ↓  
[ MediaPipe → Detectar manos ]  
    ↓  
[ Por cada mano ]:  
    ↳ Verifica si dedos están "doblados"  
    ↳ Si sí → reproducir sonido correspondiente  
    ↓  
[ Mostrar en ventana ]  
    ↓  
[ Salir con 'q' ]
```