



TECLADO VIRTUAL POR GESTOS

PROYECTO DE IA

Universidad Veracruzana

Fecha: 13/06/2025

Mendez Moreno Marco Antonio Integrante: Marco Antonio

Mendez Moreno

zS22004518@estudiantes.uv.mx

1. Introducción

Planteamiento del Problema

Las personas con discapacidad motriz parcial o severa a menudo enfrentan barreras significativas para interactuar con dispositivos electrónicos y comunicarse eficazmente con su entorno. La dependencia de interfaces tradicionales como el teclado y el ratón les impide acceder a información, expresar sus necesidades y participar plenamente en la sociedad digital. Si bien existen diferentes soluciones, estas suelen ser costosas, complejas o limitadas en su funcionalidad. La visión por computadora, en particular el seguimiento ocular (eye tracking), ha demostrado ser una vía prometedora para la comunicación alternativa. Al traducir los movimientos oculares y el parpadeo en comandos de interfaz, se puede ofrecer a estas personas una forma intuitiva y natural de interactuar con la tecnología.

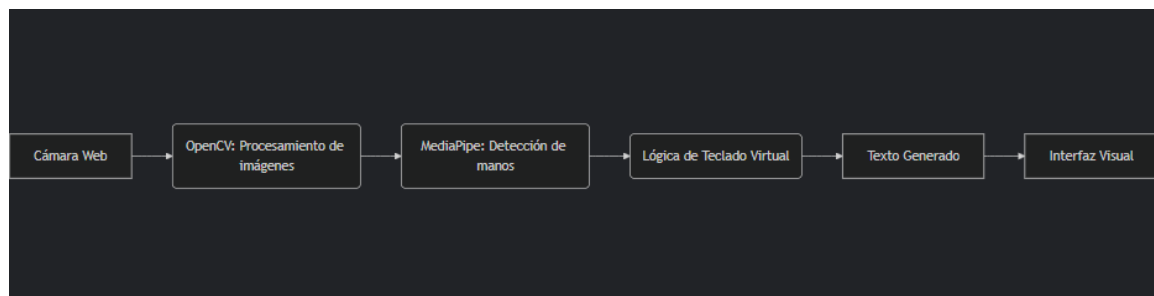
Justificación

Este proyecto propone un sistema de entrada de texto basado en seguimiento ocular y gestos faciales, utilizando tecnologías de visión por computadora de bajo costo y código abierto. Esta solución tiene como objetivo brindar una alternativa accesible, sin requerir hardware especializado, mejorando así la calidad de vida de las personas con movilidad reducida.

2. Arquitectura del Sistema

El sistema se basa en una arquitectura modular que utiliza una cámara web para capturar imágenes en tiempo real, las cuales son procesadas mediante OpenCV y MediaPipe para detectar gestos oculares y faciales. Los eventos detectados se traducen en acciones sobre un teclado virtual representado en pantalla.

Diagrama de Tecnología:



Tecnologías implementadas:

- Python 3.11
- OpenCV: procesamiento de imágenes y video
- MediaPipe: detección de landmarks faciales
- PyGame: retroalimentación visual y auditiva

3. Desarrollo y Prototipo

El prototipo incluye un teclado virtual interactivo controlado por el seguimiento ocular. Los principales componentes del sistema son:

1. Teclado Virtual: Interfaz gráfica con teclas alfanuméricas y funciones especiales como borrar o mayúsculas.
2. Seguimiento Ocular: Detección de posición de ojos y parpadeos para mover el cursor y seleccionar teclas.
3. Retroalimentación: Iluminación de teclas seleccionadas y sonidos para confirmar la selección.

4. Implementación

A continuación, se describen una función clave del sistema implementado en Python:

--- BUCLE PRINCIPAL ---

while cap.isOpened() and not exit_program:

 success, frame = cap.read()

 if not success:

 print("Ignorando fotograma vacío de la cámara.")

 continue

 frame = cv2.flip(frame, 1)

 h, w, _ = frame.shape

 image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

 image.flags.writeable = False

 results = hands.process(image)

 image.flags.writeable = True

 selected_key_info = None

 if results.multi_hand_landmarks:

 for hand_landmarks in results.multi_hand_landmarks:

 mp_drawing.draw_landmarks(frame, hand_landmarks,
mp_hands.HAND_CONNECTIONS)

```
index_tip =
hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_TIP]

cursor_x, cursor_y = int(index_tip.x * w), int(index_tip.y * h)

# Cursor

cv2.circle(frame, (cursor_x, cursor_y), 10, (0, 255, 255), -1)

cv2.circle(frame, (cursor_x, cursor_y), 12, (255, 255, 255), 2)

key_info = get_key_from_pos(cursor_x, cursor_y)

if key_info:

    selected_key_info = key_info

    if is_click(hand_landmarks):

        current_time = time.time()

        if current_time - last_click_time > click_cooldown:

            process_key(key_info)

            last_click_time = current_time

            cv2.circle(frame, (cursor_x, cursor_y), 15, (0, 0, 255), 2)

draw_keyboard(frame)

cursor_counter = (cursor_counter + 1) % 40

cv2.imshow('Teclado Virtual Centrado', frame)

if cv2.waitKey(5) & 0xFF == 27 or exit_program:

    break
```

5. Pruebas

Se realizaron pruebas para validar el funcionamiento del sistema con diferentes usuarios. Algunos casos de prueba incluyen:

Caso de Prueba	Entrada	Resultado Esperado
Selección de caracteres	Movimiento mano a la 'A'	'A' aparece en área de texto
Borrado completo	Gestos en 'BORRAR'	Texto se limpia
Alternar mayúsculas	Clic en 'MAYUS'	Teclas cambian a mayúsculas

6. Conclusiones

El prototipo demuestra la viabilidad de utilizar visión por computadora para habilitar interfaces inclusivas para personas con discapacidad motriz.

Limitaciones:

- Requiere condiciones de iluminación estables.
- Puede haber latencia en computadoras con bajo rendimiento.

Mejoras Futuras:

- Integrar modelo de predicción de palabras.
- Añadir soporte para seguimiento ocular con dispositivos móviles.
- Evaluación con usuarios reales para mejorar la usabilidad.