

Nuevos Paradigmas de Interacción



UNIVERSIDAD DE GRANADA

Memoria Técnica Gestual

Sergio Jesús Jerez Vázquez

Carlos Lao Vizcaíno

Eduardo Muñoz del Pino

Pablo Rodríguez Fernández

Índice

Índice	2
1. Introducción	3
2. Diagrama de flujo	3
3. Interfaz gráfica	4
4. Detección de gestos	5
4.1 Descripción del código	5
4.2 Gestos del frame principal	6
4.2.1 Acceso al Comedor	6
4.2.2 Parking	7
4.2.3 Actividades	8
4.3 Gestos del frame comedor	8
4.3.1 Acceso a Primeros	8
4.3.2 Acceso a Segundos	9
4.3.3 Acceso a Postres	10
4.3.4 Acceso a Menú diario	11
4.3.4.1 Acceso a Pago	12
4.4 Gestos del frame actividades	12
4.4.1 Deslizarse verticalmente por la pantalla	12
4.5 Gestos generales	13
4.5.1 Volver atrás	13
5. Bibliografía	14

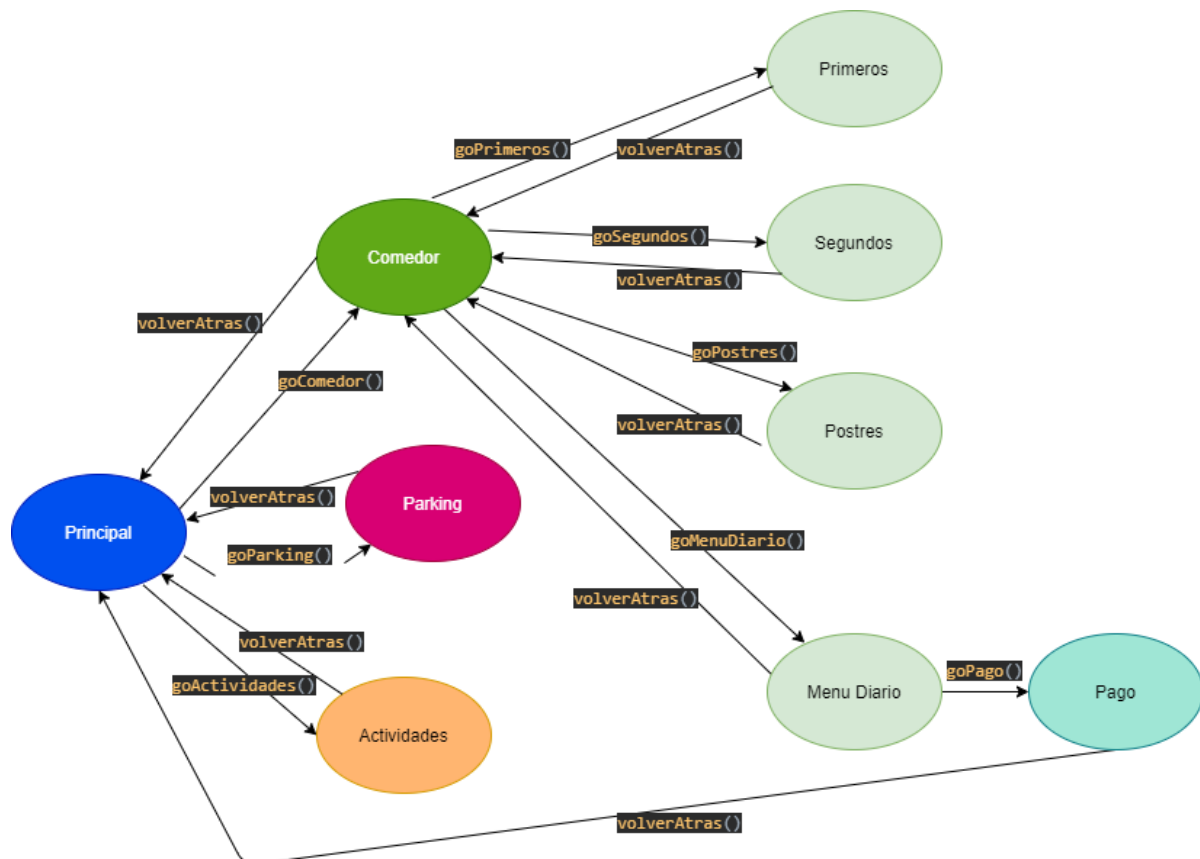
1. Introducción

En este documento se va a presentar como está hecha internamente la aplicación para escritorio de *ETSIIT Utilities*. Dicha aplicación está pensada para funcionar como un punto de información en la entrada de la escuela, el cual será controlado por gestos (sin usar la habitual pantalla táctil de este tipo de puntos).

Debido a que está pensado para estar fijo en una ubicación determinada, las funcionalidades que proporciona no son exactamente las mismas que las de la versión móvil. En este caso, el punto de información proporciona las siguientes funciones:

- **Parking:** Permite ver el número de plazas libres (y en la versión final, ver los diferentes aparcamientos a través de las cámaras instaladas para tal fin).
- **Comedor:** Permite ver la oferta global de primeros, segundos y postres. Además, permite consultar el menú del día actual y permitiría pagar dicho menú a través de una TPV instalada en el punto de información.
- **Actividades:** Permite consultar las actividades semanales programadas en la escuela, de manera similar al correo que envía semanalmente la dirección de la escuela a todos los alumnos.

2. Diagrama de flujo



3. Interfaz gráfica

Para la interfaz gráfica se ha hecho uso del paquete de Python *Tkinter*, el cual permite elaborar GUIs de manera relativamente sencilla.

Al iniciar el programa, se crea una ventana principal sobre la que se van mostrando y ocultando diferentes frames conforme se van realizando gestos. Todos los frames se crean al ejecutar la aplicación pero solo se muestra el *framePrincipal* de inicio. Cada frame contiene el texto e imágenes correspondiente a cada sección de la aplicación

Para saber en qué frame estamos hacemos uso de variables booleanas de control. Estas se comprueban a la hora de detectar el gesto y se modifican en los siguientes métodos:

```
goPrincipal()
goParking()
goActividades()
goComedor()
goPrimeros()
goSegundos()
goPostres()
goMenuDiario()
goPago()
volverAtras()
```

Estos son invocados cuando se detecta el correspondiente gesto y lo que hacen es mostrar y ocultar los frames que corresponda en cada caso. Cada uno de los métodos anteriores corresponden a uno de los frames de la aplicación excepto el último (*volverAtras*) que sirve para volver hacia atrás siguiendo el flujo ya visto en el diagrama.

En este último se comprueba en qué frame estamos, usando las variables de control y se invoca el método del frame que corresponda

```
mover(inclinacion)
```

En el caso del fram actividades hay un gesto para hacer scroll en el frame. Dicho gesto da valores positivos o negativos, por ello, el método *volver(inclinación)* el cual invoca, comprueba si es positivo o negativo y sube o baja el scroll según convenga.

4. Detección de gestos

En este apartado, en primer lugar, se describen brevemente los elementos proporcionados por el controlador de *Leap* utilizados para reconocer los distintos gestos diseñados. Tras esto, se muestran los diferentes gestos utilizados para interaccionar con la aplicación, indicando cómo realizarlos y comentando brevemente las comprobaciones que se realizan internamente para detectarlos.

Para qué sea más fácil de seguir la descripción de los gestos, éstos se agrupan en función del frame en el que son reconocidos y en función de la acción que realicen.

4.1 Descripción del código

La mayoría de los gestos se basan en el control del número de dedos que tiene extendidos el usuario o el movimiento de ellos. Es posible obtener la información individual sobre cada dedo a través del atributo *hand* del *controller.frame()*:

```
frame = controller.frame()
for hand in frame.hands:
    #Obtenemos los distintos dedos de la mano
    thumb = hand.fingers[0]
    index = hand.fingers[1]
    middle = hand.fingers[2]
    ring = hand.fingers[3]
    little = hand.fingers[4]
```

Se realiza un bucle *for* por si en un mismo frame se detectan ambas manos (derecha e izquierda).

Cada dedo tiene un atributo que indica si está extendido o no (*is_extended*). A través de este atributo se puede comprobar, por ejemplo, que solo está extendido el dedo índice (*index*):

```
if index.is_extended and not(middle.is_extended) and not(thumb.is_extended)
and not(ring.is_extended) and not(little.is_extended)
```

En ocasiones también es necesario comprobar si la mano está quieta (o prácticamente quieta) o si existe cierto movimiento. Esto se puede lograr empleando el atributo *palm_velocity*, también de la clase *hand*. Por ejemplo, para comprobar si la mano está quieta, se analiza si la velocidad de la palma de la mano en los tres

ejes es menor que cierto umbral (para aceptar el gesto aunque exista cierto temblor en las manos):

```
abs(hand.palm_velocity[1]) < 50 and abs(hand.palm_velocity[0]) < 50 and
abs(hand.palm_velocity[2]) < 50)
```

Uno de los gestos necesita saber la inclinación de la palma de la mano respecto al plano horizontal. Esto es posible obtenerlo también mediante un atributo de la clase *hand*:

```
direction = hand.direction
```

Finalmente, también se hace uso del gesto del círculo, el cuál es uno de los gestos predefinidos por la propia librería. La manera de comprobar este gesto es diferente al resto, ya que los gestos predefinidos se almacenan directamente en el propio frame, y se debe comprobar mediante un bucle cuál de ellos se ha detectado (de manera similar al bucle de detección para las manos derecha e izquierda):

```
for gesture in frame.gestures():
    #Volver atras
    if gesture.type == Leap.Gesture.TYPE_CIRCLE
```

4.2 Gestos del frame principal

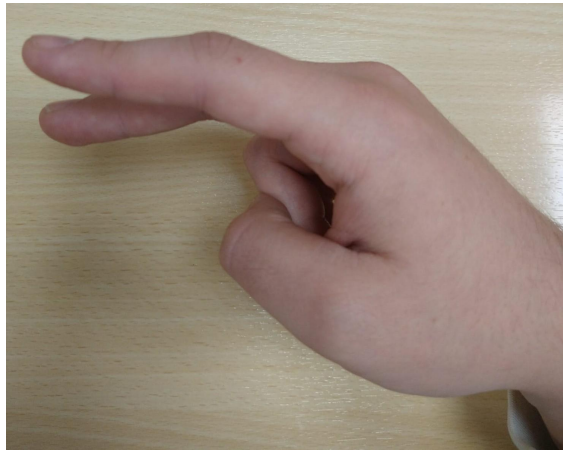
4.2.1 Acceso al Comedor

Descripción del gesto

El gesto para acceder a la ventana se realiza con los dedos índice y corazón de la mano derecha, moviéndolos de derecha a izquierda, imitando el movimiento de llevarse la comida a la boca con una cuchara o tenedor.



Posición inicial



Posición final

Implementación

Se comprueba que los únicos dedos extendidos sean el índice y el corazón, y que además la aceleración de la palma de la mano sea mayor que ciertos umbrales (obtenidos mediante pruebas), ya que si no se podría entrar en esta pantalla simplemente estirando los dos dedos (sin realizar ningún movimiento).

4.2.2 Parking

Descripción del gesto

El gesto para acceder a la venta parking se realiza con el puño cerrado y haciendo como que se arranca una moto. También se comprueba la velocidad de la mano



Posición inicial



Posición final

Implementación

Para este gesto se comprueba que ninguno de los dedos está extendido, es decir, el puño está cerrado. También se comprueba que la velocidad vertical sea alta y las demás muy bajas y que la inclinación vertical (cabeceo) de la mano sea menor de 90 grados en valor absoluto. De esta manera el gesto se asemeja a arrancar una moto

4.2.3 Actividades

Descripción del gesto

El gesto para acceder a la ventana de las actividades consiste en tener abierta la mano con todos los dedos extendidos.



Implementación

Para este gesto simplemente se comprueba si todos los dedos están extendidos y si la velocidad de la palma de la mano en cualquier dimensión es menor que un umbral. La restricción de velocidad es necesaria para detectar que la mano está quieta.

4.3 Gestos del frame comedor

4.3.1 Acceso a Primeros

Descripción del gesto

El gesto para acceder a la ventana se realiza con el dedo índice desplegado y el resto plegados, imitando la representación natural del número uno.



Implementación

Se comprueba que el único dedo extendido sea el índice. Además, se comprueba que la aceleración de la mano sea mínima. Esto es debido a que, de no hacerse esta comprobación, este gesto puede entrar en conflicto con otros que utilizan el mismo dedo pero en movimiento (concretamente, el círculo).

4.3.2 Acceso a Segundos

Descripción del gesto

El gesto para acceder a la ventana se realiza con los dedos índice y corazón desplegados y el resto plegados, imitando la representación natural del número dos.



Implementación

Se comprueba que los únicos dedos extendidos sean el índice y el corazón. Además, se comprueba que la aceleración de la mano sea mínima. Esto es debido a que, de no hacerse esta comprobación, este gesto puede entrar en conflicto con el gesto de entrar en la ventana del *Comedor*. Dicho gesto utiliza los mismos dedos pero en movimiento, con lo cual, es muy posible que nada más entrar en la ventana, el Leap detecte otra vez los mismo dedos desde la nueva pantalla, provocando que se entre directamente desde la pantalla principal a la de segundos platos.

4.3.3 Acceso a Postres

Descripción del gesto

El gesto para acceder a la ventana se realiza levantando exclusivamente 3 dedos, para imitar al número tres. Sin embargo, en este caso no existe una combinación de dedos “estándar” como existe con el uno y con el dos. Por esta razón, se admiten las siguientes combinaciones:

1. Pulgar, índice y corazón.
2. Índice, corazón y anular.
3. Corazón, anular y meñique.



Posición 1



Posición 2



Posición 3

Implementación

Se comprueba que los únicos dedos extendidos sean algunos de los siguientes:

- Pulgar, índice y corazón.
- Índice, corazón y anular.

- Corazón, anular y meñique.

Como en los casos anteriores, también se comprueba que la mano está relativamente quieta.

4.3.4 Acceso a Menú diario

Descripción del gesto

El gesto para acceder a la ventana se realiza extendiendo los cinco dedos de la mano.



Implementación

Se comprueba que todos los dedos de la mano están extendidos.

4.3.4.1 Acceso a Pago

Descripción del gesto

Para pasar a la pantalla de pago hay que realizar el gesto de levantar el pulgar.



Implementación

Se comprueba que el pulgar está levantado y el resto de dedos están cerrados. Permite hacer el gesto horizontalmente como verticalmente, aunque de la segunda forma no siempre es detectado.

4.4 Gestos del frame actividades

4.4.1 Deslizarse verticalmente por la pantalla

Descripción del gesto

El gesto para hacer “scroll” se basa en inclinar la palma abierta de la mano con los dedos apuntando hacia arriba si se quiere subir y hacia abajo si se desea bajar.



Subir por la pantalla



Bajar por la pantalla

Implementación

En este gesto comprobamos que todos los dedos menos el pulgar están extendidos (el pulgar puede estar cerrado o extendido) y que la dirección de la palma hacia los dedos se encuentra en un ángulo mayor que un cierto umbral respecto de si

estuviera la palma extendida de manera totalmente horizontal sobre el Leap Motion. Para ello se usa el atributo de Leap `direction.pitch`, el cuál nos dice el ángulo en el eje Y. Dicho atributo se transforma de radianes a grados para facilitar el trabajo.

4.5 Gestos generales

4.5.1 Volver atrás

Descripción del gesto

Este gesto es uno de los ya definidos en la librería de Leap y consiste en dibujar un círculo en el aire.



Implementación

Al gesto ya definido le hemos añadido que compruebe que el dedo índice esté extendido y que los dedos corazón, anular y meñique estén cerrados. Esta comprobación es necesaria para que no quede la posibilidad de hacer el círculo con otra disposición de los dedos que haga confundir al Leap con otro de los movimientos que hemos definido.

5. Bibliografía

<https://developer-archive.leapmotion.com/documentation/v2/python>

<https://es.stackoverflow.com/questions/195494/label-tkinter-varias-lineas>

<https://docs.hektorprofe.net/python/interfaces-graficas-con-tkinter/>