

# XXII JORNADAS EN INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

## MOUSE PARA PERSONAS CON DISCAPACIDAD MOTRIZ

Jarrín Zambrano Dina Patricia, Ing.  
Álvarez Rueda Robin, PhD.

Grupo de Aplicaciones en Biotecnología y Telecuidado (GABT)

Escuela Politécnica Nacional

### INTRODUCCION

El presente proyecto se desarrolla con el propósito de permitir a personas con discapacidad motriz, tener la posibilidad de utilizar el mouse del computador.

Ya que se utilizara una cámara web de bajo costo, para evitar problemas de iluminación, se utilizará un objeto reflector que facilite su detección mediante el procesamiento digital de imágenes, de modo que su seguimiento emule el puntero del mouse. Se estudiará alternativas para la activación tanto del botón derecho como del izquierdo.

Finalmente se realizan las pruebas de los diferentes algoritmos planteados, para definir cuál de ellos es el más idóneo que permitan cumplir con los objetivos planteados.

### 1. ESTADO DEL ARTE

En esta sección se van a presentar avances y técnicas que se han desarrollado hasta el momento; los cuales son comentados a continuación:

**1.1 Camera Mouse** realizado por la Universidad “Boston College”; emulado por detección facial, utilizan una cámara web con conexión USB, haciendo de puntero la nariz del usuario, o el borde de las cejas de la imagen captada, el control del mouse se dará por el movimiento de la cabeza. [1]

Este trabajo ha sido dedicado para adolescentes con diferentes discapacidades que impiden movimientos motrices normales. En la figura1 se puede ver un ejemplo de aplicación de este sistema.



**Figura1.** Zubair, estudiante del colegio “Holly Bank School” usando el sistema de la cámara web. [5]

**1.2 Emulador de Mouse para personas con alteraciones motoras:** realizado en la escuela ORT de Argentina con el apoyo de Centro de Investigación y Desarrollo de Asistencias Tecnológicas para la Discapacidad del Instituto Nacional de Tecnología Industrial (INTI). [2]

Utilizan una cámara web ubicada en la cabeza del usuario, sujeta con una bincha, gorro o casco y con un led infrarrojo ubicado sobre el monitor. El clic izquierdo, derecho y doble clic lo hacen mediante software.

A continuación se muestra en la figura2 esta aplicación.

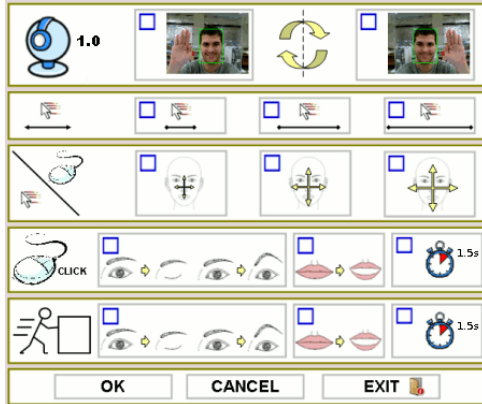


**Figura2.** Mouse para personas con alteraciones motoras.

**1.3 HeadMouse1** realizado por el Grupo de Robótica - Universitat de Lleida.

Utilizan una cámara web y por medio del movimiento de la cabeza conserva el control del puntero y las diferentes opciones de clic. [3]

A continuación se mostrará en la figura3 un menú de diferentes decisiones para un óptimo control del puntero del mouse.



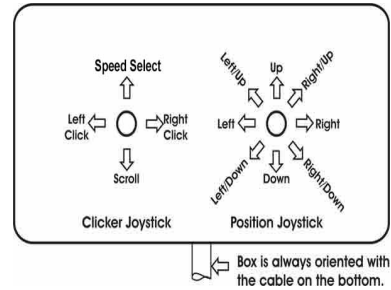
**Figura3.** Menú de opciones de aplicaciones.

**1.4 Tetramouse** realizado por TetraLite Products, Seattle WA, USA, con la boca realiza el control emulando el clic derecho y el clic izquierdo, este dispositivo está conectado al computador por USB o con el adaptador PS/2. [4].



**Figura6.** Paquete de Tetramouse

A continuación en la figura7 se muestra el cuadro de dialogo de aplicaciones para el puntero del mouse.



**Figura7.** Funciones del Tetramouse

## 2. MOUSE PARA PERSONAS CON DISCAPACIDAD MOTRIZ



**Figura8.** Mouse para personas con discapacidad motriz.

Después de un breve análisis en comparación de otros procedimientos se podrá notar ventajas y desventajas acerca del control del puntero del mouse.

El uso del objeto reflector ayudará a mantener el control del puntero del mouse, lo que resulta más óptimo en comparación otros sistemas dados por mouse facial, ya que depende mucho más de la iluminación del ambiente.

Como se habrá notado otros paquetes (software de instalación, cámara web y dispositivos extras), tienen un costo considerable, mientras que nuestro programa, aún en desarrollo.

### 2.1 MATERIAL Y MÉTODOS

#### a) MATERIAL:

##### Hardware:

- Cámara Web (de características básicas).

- Un objeto reflector.
- Un computador.

**Software:**

- Software desarrollado.
- Sistema Operativo XP o Windows Vista.

**b) METODO:**

El algoritmo que se ha desarrollado es el resumido en la figura8 y que produjo los siguientes resultados:

**3. RESULTADOS**

Se toma una muestra de la imagen, capturada por la webcam que va a permitir el inicio del análisis (figura9).



**Figura9.** Adquisición de imagen

La imagen de colores verdaderos RGB se convierte a escala de grises para facilitar la aplicación de los algoritmos posteriores (figura10).



**Figura10.** Imagen en escala de Grises

La imagen de muestra RGB, puede ser de precisión uint8, uint16, solo, o doble. Al ser convertida a imagen en escala de grises, esta es de la misma clase como la imagen RGB. Para asegurar los mejores resultados, convertimos a la imagen a precisión doble, como se puede ver en la figura11.



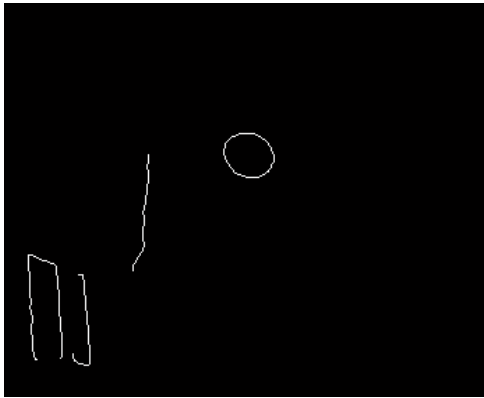
**Figura11.** Imagen doble

Sobre esta imagen, aplicamos un algoritmo de detección de bordes, mediante el algoritmo de Prewitt (figura12).



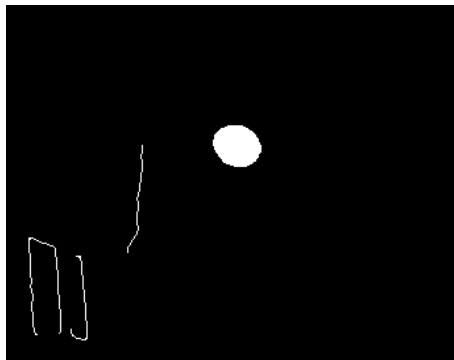
**Figura12.** Detección de bordes mediante algoritmo de PREWIT

A la imagen resultante, le aplicamos un algoritmo de eliminación de objetos inferiores a 80 pixeles (figura 13).



**Figura13.** Imagen después de aplicar el algoritmo de eliminación de objetos inferiores a 80 pixeles.

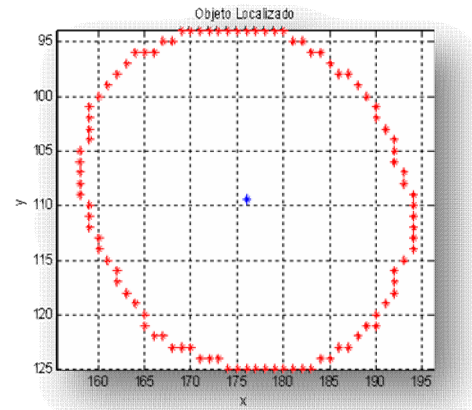
Mediante una operación morfológica cerrada, se rellena los objetos cerrados de la imagen anterior (figura14).



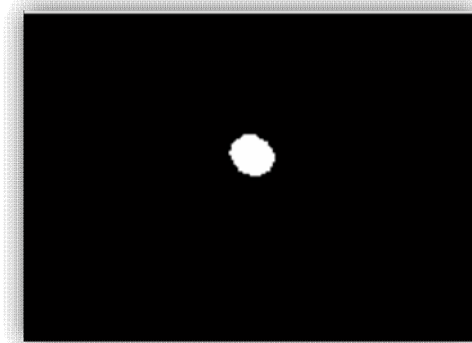
**Figura14.** Relleno de objetos cerrados mediante operación morfológica cerrada.

A partir de este resultado, procedemos a realizar segmentación para poder reconocer objetos y cuestionar:  
 ¿Cuántos objetos hay?  
 ¿Cuál de ellos es redondeado?

Para responder a las preguntas anteriores, se extrae las filas y columnas de cada objeto, ubicamos su **centroide** (figura15) y respecto de este, se compara la longitud del eje vertical con la del eje horizontal aceptando una cierta tolerancia con lo cual estaremos verificando si fue o no un objeto redondeado (figura16).

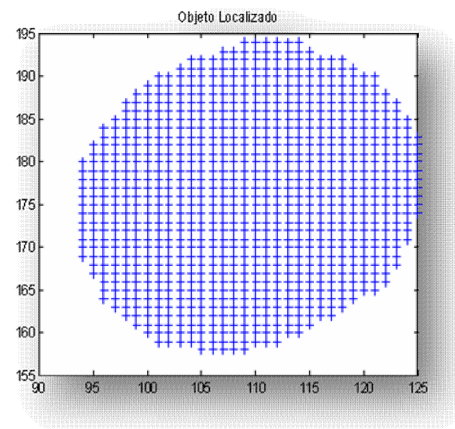


**Figura15.** Obtención del centroide del objeto seleccionado.



**Figura16.** Resultado de aplicar el algoritmo de detección de objetos redondeados.

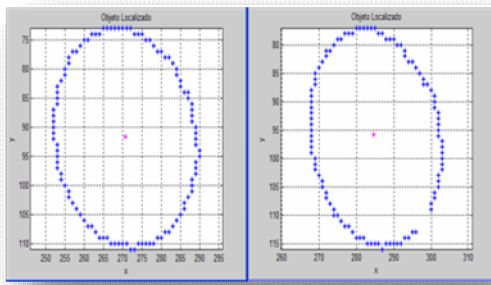
Para facilitar la visualización de filas y columnas del objeto, y así verificar el resultado del algoritmo anterior, se utiliza una herramienta de matlab (figura 17).



**Figura17.** Visualización de la ubicación del objeto en pixeles

Finalmente, tomando solamente dicho centroide, este será el que sirva para seguimiento del puntero del mouse (figura

18).

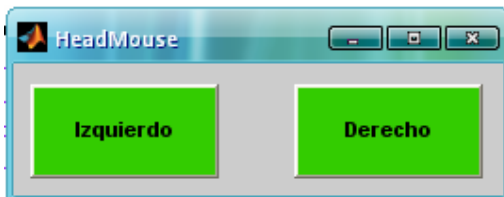


**Figura18.** Centroide que sirve para seguimiento del puntero: dos imágenes en posiciones diferentes.

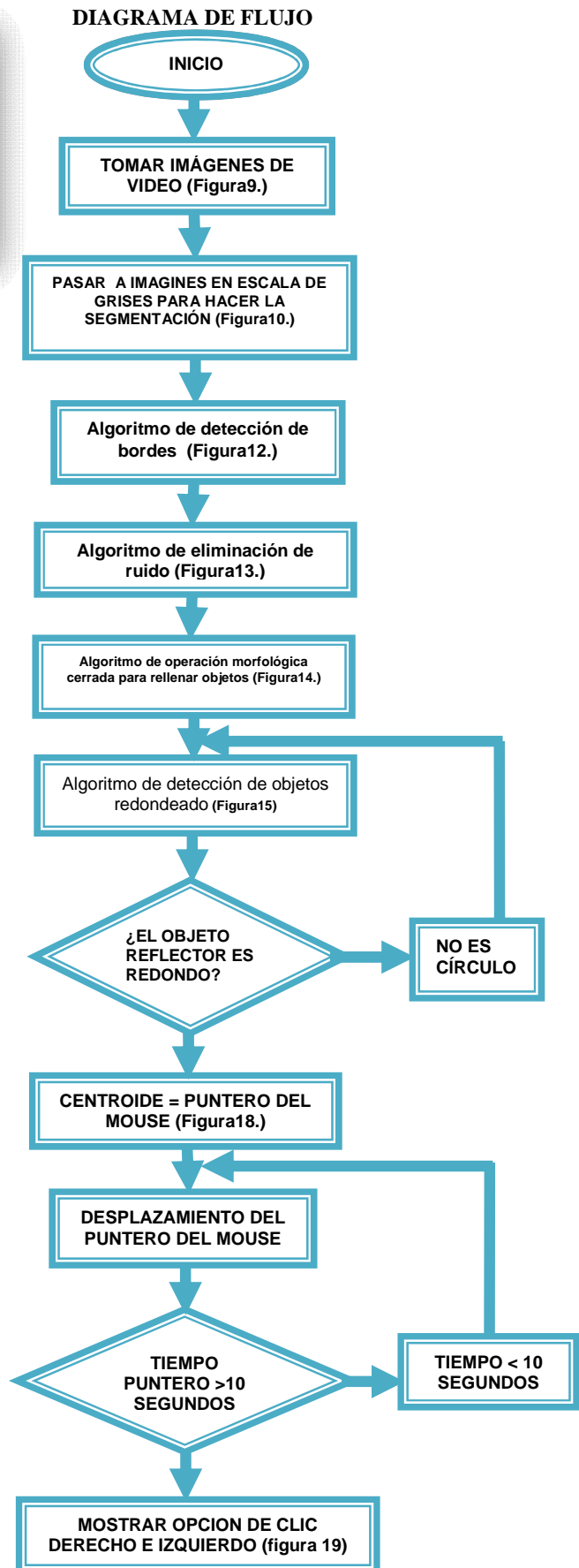
### 3.1 Método para seleccionar clic derecho o clic izquierdo.

Se procede a leer el tiempo que el usuario mantiene inmóvil su cabeza (el puntero del mouse) en un lugar determinado y si el tiempo es mayor a lo indicado por el programa (por ejemplo  $t > 10s$ ), se ejecutará el cuadro de selección de clic derecho y clic izquierdo (figura 19).

Para la utilización de los comandos del puntero del mouse se utiliza las librerías de Java y Robot, con el fin de hacer coincidir el puntero del mouse con el centroide del objeto calculado.



**Figura 19.** Opcion de clic derecho y clic izquierdo.



#### 4. DISCUSION

- Al igual que proyectos anteriores, se ha buscado la mejor alternativa para que el proyecto sea más accesible para personas con discapacidad motriz y que puedan hacer uso de este tipo de tecnología, viendo la factibilidad de uso y alcance para el usuario.
- El proyecto actual se está implementando por primera vez en nuestro país lo que es una gran iniciativa para que las personas con discapacidad motriz puedan hacer uso del mouse.
- Haciendo un breve análisis de los problemas de iluminación que se dan en otros proyectos similares, como el de detección facial, se determinó que el uso del objeto reflector servirá para minimizar este inconveniente.
- Se probaron varios métodos para eliminar objetos no deseados (ruido), para esto se intentó con Filtros en 2D prediseñados, con métodos basados en pixeles (umbralización) pero ninguno dio los resultados esperados. Entonces la mejor solución que se encontró fue la eliminación de pixeles “suaves” por medio de la herramienta que el mismo toolbox de Matlab posee, y finalmente se eliminaron los objetos no deseados.
- Ahora en base a las características del objeto se extrae el valor del centroide y se lo hace coincidir con el puntero del mouse, mediante librerías de Java y Robot se ejecuta el control del puntero del mouse en Matlab, logrando así el control del desplazamiento de este en la pantalla, con ayuda del movimiento de la cabeza, pero siempre tratando de que el haz de luz incida en el objeto reflector y la cámara web detecte el reflejo.
- Analizando aspectos económicos de otros proyectos se ha considerado que nuestro proyecto es más económico debido a que simplemente se usa una cámara web con requerimientos básicos y un objeto reflector como un espejo redondo de radio pequeño y fácil de encontrar en el mercado, haciendo así que el requerimiento presupuestario sea mínimo en comparación a otros proyectos.

#### 5. CONCLUSIONES

- El programa utilizado para el desarrollo del proyecto es Matlab, el mismo que necesita un alto nivel de programación y con la ayuda de las librerías de Java y Robot se pudo desarrollar la emulación del movimiento del mouse.
- Es aconsejable revisar las propiedades de la cámara web como es la resolución que esta ofrece y si fuera necesario ajustar las propiedades de brillo, el contraste, los balances de colores, etc.
- Las imágenes obtenidas deben ser procesadas con el propósito de facilitar la detección y discriminación de objetos que se presentan en el plano de la imagen.
- Para el procesamiento de las imágenes es necesario realizar operaciones morfológicas de relleno, filtrado y eliminación de ruido, con el propósito de poder optimizar la detección de nuestro objeto base
- Es necesario en el proceso de detección de imágenes obtener las propiedades de cada elemento con el fin de poder seleccionar el elemento deseado que se encuentran en el mismo plano de la imagen.
- Finalmente se hizo un análisis del puntero del mouse respecto al cuadro de selección de clic derecho y clic izquierdo, para lograr una concordancia de funcionamiento entre el cuadro de selección y el puntero.

#### 6. REFERENCIAS BIBLIOGRAFICAS

- [1]<http://www.bc.edu/schools/csom/eagleeyes/cameramouse/about.html#q1>
- [2]<http://www.canalar.com.ar/noticias/noticiamuestra.asp?Id=4496#comentario>
- [3]<http://robotica.udl.cat/headmouse/headmouse1/headmouse1.html#CONTROL>
- [4]<http://tetramouse.com/manual.html#connect>
- [5]<http://www.bc.edu/schools/csom/eagleeyes/cameramouse/gallery/page11.html>

## 7. BIOGRAFÍAS

### Dina P. Jarrín Zambrano



Nació en Quito Ecuador el 14 de mayo de 1980, se encuentra cursando el último semestre de la carrera de Ingeniería Electrónica Control y Automatización en la facultad de Ingeniería Eléctrica y Electrónica. Fue miembro activo de la IEEE, participó en eventos realizados en su Universidad en el área de Electrónica y Potencia, Instrumentación Industrial, además fue parte del grupo de Robótica de la facultad, el mismo que ganó el primer lugar en el concurso internacional realizado en Perú.

### Robin Álvarez Rueda



Nacido en Cayambe, Ecuador, en 1969. Ingeniero en Telecomunicaciones, graduado de la Escuela Politécnica Nacional, Quito - Ecuador, 1995; M.Sc. en Telecomunicaciones por la Universidad de Cantabria, Santander - España, 2001; Ph.D en Telecomunicaciones por la Universidad Politécnica de Madrid-España, Enero-2006. Su interés desde hace seis años está en el campo de la bioingeniería.