

Introducción:

El uso de la electromiografía es frecuentemente relacionado con análisis cerebrales, musculares, uso de brazos robóticos, terapias, etc.. Pero siempre dentro del campo de la medicina por lo que se suele empatizar con estos términos con cierto temor. La habilidad de brindar juegos de entrenamiento y que a su vez sean divertidos es de vital importancia [1] para comenzar a adquirir una mentalidad en la sociedad más familiarizada y cómoda con la electromiografía ya que, al fin y al cabo, en las próximas décadas los desarrollos y aplicaciones futuras van en esta dirección, la bio-robótica.

Desafortunadamente, los próstéticos operados por EMG son percibidos como difíciles de operar, por lo que causa que muchos usuarios potenciales opten por llevar próstéticos mucho más simples o ninguno. [2]

Requisitos:

- Crear unos controles en el videojuego que sean binarios para simplificar el análisis de datos del EMG ya que los controles del usuario serán recibidos por serial de forma binaria (es decir, comandos simples como arriba, abajo, derecha, izquierda).
- Accesibilidad y curva de aprendizaje del videojuego debe ser fácil, de manera que no se requiera una alta capacidad de adaptación para poder acostumbrarse rápidamente al videojuego.
- Adaptación a un público genérico.
- Juegos competitivos, ya que estudios [1] sugieren que un juego no competitivo y sin niveles, al ser completada, se convierte en una tarea repetitiva y monótona.

Propuesta:

Con la gran variedad de modalidades de juego, hemos de seleccionar alguna que permita varios niveles de dificultad, se ha pensado en realizar un pequeño juego en el que se atrapan patos con una caña de pescar, juego muy utilizado en las ferias de los pueblos de España. Pero esto no permite niveles de dificultad mas allá de lo que el usuario tiene en la curva de aprendizaje, una vez lo ha superado, el modo de juego se convierte en monótono y repetitivo.

Una alternativa es un juego de habilidad con alguna herramienta en la que el usuario tenga que atrapar (de manera más sutil) algún objeto en movimiento, que permita la diferenciación de objetivos que puntúan y otros que penalizan. Por ejemplo, el usuario sujeta una espada y trata de cortar frutas en movimiento.

Para añadir de alguna forma cierta aleatoriedad (y, por lo tanto, eliminación de monotonía) y varios grados de dificultad, se puede incluir una zona estática donde se indica la fruta actual que puede puntuar, de tal manera que, si se daña una fruta distinta, no puntúa. La misma situación ocurre con aquellas que penalizan. Aunque se pueden definir algunos objetos preseleccionados en las primeras etapas de desarrollo del videojuego.

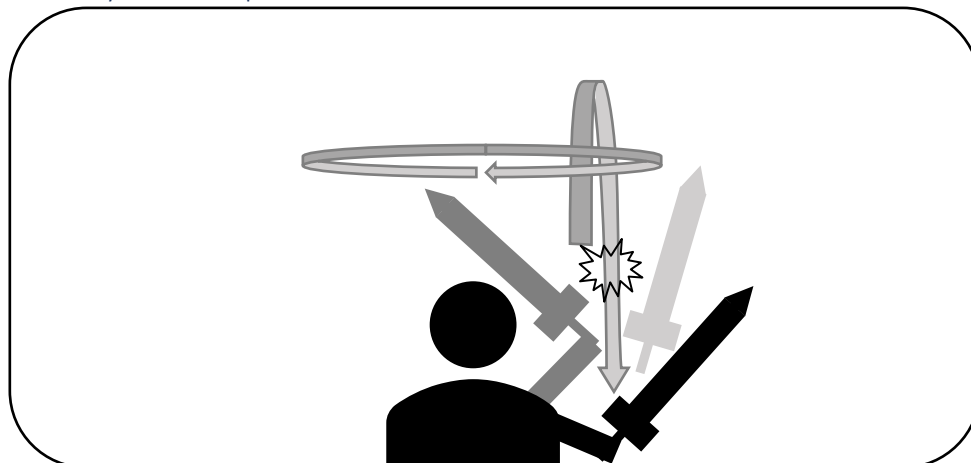
Interfaz Gráfica:

Composición del escenario estático:



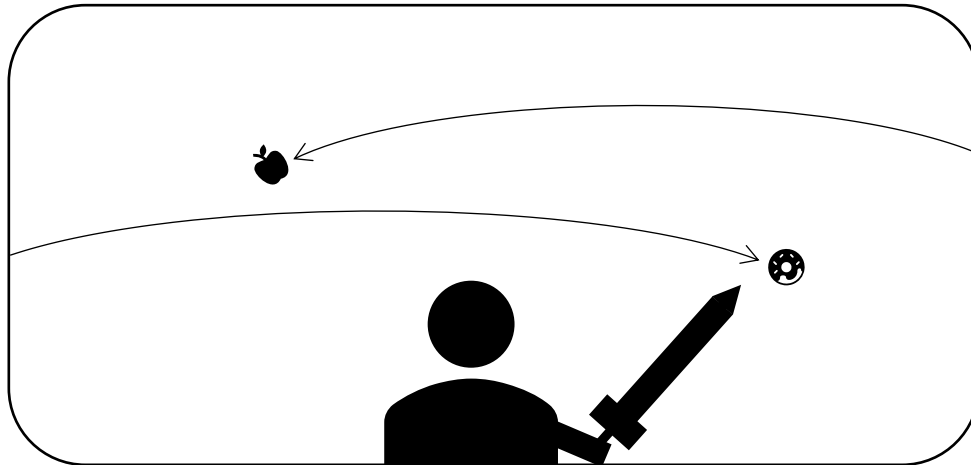
- Objetivo: Indicador del objetivo, podrá ser cualquier objeto que se haya diseñado. Si el jugador "atina" uno de estos objetivos en movimiento, se incrementa la puntuación.
- Puntuación: Incrementa o decrementa dependiendo de los objetos "atinados".
- Reparación derecha/izquierda: Los objetos pueden reaparecer desde la derecha hacia izquierda o desde izquierda hacia derecha.

Movimientos y acciones posibles del usuario:



- El usuario puede girar el avatar de manera leve para dirigir su espada levemente hacia izquierda o derecha.
- Al momento de aplicar fuerza con la espada se produce un golpe de espada. Este golpe se realiza en dirección vertical y realizará un corte tajante a cualquier objeto que se encuentre en esta zona.

Movimientos de los objetos:



- Los objetos, tanto los objetivos que puntúan como los que penalizan aparecerán de en la izquierda o derecha de manera aleatoria en sentido contrario.
- Los objetos describirán una parábola producida por la gravedad, como si de un objeto lanzado desde los extremos hacia el centro se tratase.

Especificaciones técnicas:

Conexión con hardware:

Deberá de realizarse de manera automática. Con un método de sincronización de manera que el juego pueda ejecutarse de manera autónoma y al existir una conexión Serial, pueda ser utilizado. Este sistema es comúnmente llamado “plug&play”.

Entornos estáticos 2D:

Aquellos elementos que permanezcan estáticos deberán de renderizarse los últimos ya que los objetos dinámicos no pueden solapar por encima los elementos estáticos. Cabe la posibilidad de renderizar en 3D y generar los elementos estáticos en el eje $z = 0$, y los demás objetos con $z < 0$.

Fondo de pantalla:

Deberá generarse el primero, a decisión del desarrollador la imagen, que puede ser aleatoria intercambiando la imagen por cada ejecución del juego.

Avatar:

El avatar será un tipo de objeto, en el que como mínimo, tendrá almacenado su giro actual y la posición de la espada (arriba o abajo).

Objetos dinámicos:

Los objetos dinámicos será una lista de objetos de un tipo especial de objeto. Estos objetos deberán de tener como mínimo, su rotación, su vector velocidad y posición actuales. Además, podrán incluirse efectos especiales de destrucción (por ejemplo, partiéndose en dos) con la generación de nuevos objetos de otro tipo. Su reaparición (izquierda o derecha) y el ángulo de inclinación con respecto al plano deberá de ser aleatorio.

Su imagen correspondiente también deberá de ser aleatoria dentro de un conjunto de imágenes preseleccionadas.

Además, se puede añadir cierto nivel de dificultad con el paso del tiempo o pasada cierta puntuación, por ejemplo, con el ritmo de aparición de los objetos y su velocidad, que puede incrementarse de manera notable.

Panel objetivo:

En el panel objetivo, deberá mostrarse un objeto de tipo igual que los objetos dinámicos cada cierto tiempo aleatorio (entre 2 a 5 segundos, variable con el tiempo). El avatar, al realizar un corte sobre un objeto dinámico, se comparará con este objeto.

Panel puntuación:

En este panel se muestra la puntuación actual comenzando por 0. Cuando el avatar realiza un corte y coincide con el objeto actual en el panel objetivo, se incrementa el valor.

Panel de calibración:

Es probable que para las fases iniciales del proyecto se tenga que realizar algún tipo de panel para la calibración inicial del dispositivo. Para ello se podrá diseñar una interfaz para ajustar la sensibilidad del dispositivo. Utilizando la calibración del usuario como pantalla de carga previa al juego, algo parecido a lo que ocurre con los mandos wii.

Referencias:

- [1] The Impact of a Custom Electromyograph (EMG) Controller on Player Enjoyment of Games Designed to Teach the Use of Prosthetic Arms (<https://doi.org/10.1007/s40869-018-0060-0>)
- [2] (Bidiss and Chau 2007) - Bidiss, E. A., & Chau, T. T. (2007). Upper limb prosthesis use and abandonment: A survey of the last 25 years. *Prosthetics and Orthotics International*, 31 (3), 236–257.