

**TECNOLOGÍA DE VISIÓN ARTIFICIAL 3D PARA DETECTAR  
MOVIMIENTOS EN PERSONAS CON CONDICIONES MUSCULARES  
DIVERSAS A TRAVÉS DE UN APLICATIVO INFORMÁTICO****3D ARTIFICIAL VISION TECHNOLOGY FOR DETECTING  
MOVEMENTS IN PEOPLE WITH MUSCULAR DISABILITIES  
THROUGH A COMPUTER APPLICATION**

 MSc. Alejandro Marín-Cano\*,  Esp. Álvaro Romero-Acero\*,  
 PhD. Jovani Alberto Jiménez-Builes\*

\* **Universidad Nacional de Colombia**, Facultad de Minas, Grupo de investigación  
Inteligencia Artificial en Educación.  
Carrera 80 Nro. 65-223 oficina M8A-309, Medellín, Antioquia, Colombia.  
Tel.: +57 604 4255222  
E-mail: {amarincan, alromeroac, jajimen1}@unal.edu.co

**Cómo citar:** Marín Cano, A., Romero Acero, Álvaro, & Jiménez Builes, J. A. (2023). TECNOLOGÍA DE VISIÓN ARTIFICIAL 3D PARA DETECTAR MOVIMIENTOS EN PERSONAS CON DIFERENTES CONDICIONES MUSCULARES ATRAVÉS DE UN APLICATIVO INFORMÁTICO. REVISTA COLOMBIANA DE TECNOLOGÍAS DE AVANZADA (RCTA), 2(42), 115–121.  
Recovered from <https://ojs.unipamplona.edu.co/index.php/rcta/article/view/2714>

Derechos de autor 2023 Revista Colombiana de Tecnologías de Avanzada (RCTA).  
Esta obra está bajo una licencia internacional [Creative Commons Atribución-NoComercial 4.0](https://creativecommons.org/licenses/by-nc/4.0/).



**Resumen:** en este artículo, se presenta una aplicación informática que utiliza la tecnología de inteligencia artificial conocida como visión 3D. Esta aplicación ofrece una forma sencilla de permitir la interacción de personas que padecen condiciones musculares diversas con un computador. A pesar de la abundancia de dispositivos en el mercado capaces de detectar movimientos y reconocer gestos, existe una escasez de desarrollos específicos que les faciliten el acceso y uso de los medios de información y comunicación orientados a personas con limitaciones motoras. Los resultados obtenidos al utilizar esta aplicación indican que es una herramienta útil cuando se integra en un proceso de inclusión social, permitiendo a las personas con condiciones musculares diversas ingresar a entornos laborales y educativos de manera más efectiva.

**Palabras clave:** visión artificial, condiciones musculares diversas, inclusión educativa, transformación digital, inteligencia artificial.

**Abstract:** This article describes a computer program that incorporates 3D artificial vision technology, a branch of artificial intelligence. This application provides a straightforward way for individuals with various muscular conditions to interact with a computer. Despite the plethora of devices on the market capable of detecting movements and recognizing gestures, there is a shortage of innovations designed to facilitate access and use of information and communication media for people with motor limitations. The results of this application indicate that it is a valuable aid when used in a social inclusion process, allowing individuals with a variety of muscular conditions to participate more effectively in work and educational environments.

**Keywords:** artificial vision, diverse muscle conditions, educational inclusion, digital transformation, artificial intelligence.

## 1. INTRODUCCIÓN

Actualmente, La visión artificial se utiliza en una amplia gama de aplicaciones informáticas orientadas principalmente a la industria y la investigación. Estas aplicaciones incluyen, entre otras, la navegación autónoma (Villaverde, 2009), sistemas de vigilancia (Figueroa *et al.*, 2018), realidad aumentada (Bautista & Archila, 2011), e inclusión social (Mauri, 2004).

De acuerdo con Huynh *et al.* (2023), los dispositivos basados en la visión artificial permiten la interacción con el ordenador a través de diversas técnicas; dos de ellas son: (i) Movimiento ocular: en esta categoría, existen diversos desarrollos, incluyendo el BET2.0, el cual permite al usuario controlar el cursor del ordenador mediante un dispositivo basado en visión artificial y software instalado en un ordenador personal. Esta aplicación beneficia principalmente a aquellas personas con limitaciones de movilidad severa que no pueden usar periféricos convencionales como un ratón y un teclado (2023, Jones *et al.*). (ii) Detección de movimientos con la mano y gestos: esta categoría comprende tanto el reconocimiento del movimiento del cuerpo como el reconocimiento de gestos con la mano. Este tipo de software está basado en la detección de movimiento a través de la visión estéreo, que se puede clasificar en dos grupos (Chen *et al.*, 2023): detección basada en modelos de mano y brazo, y detección de gestos de mano (posiciones de los dedos, orientación, etc.) (Song & Demirdjian, 2010). Para este tipo de técnicas hay diversos dispositivos. For this type of technique, there are several devices in the mercado, incluyendo the Kinect, iSense™3D scanner, y the Xtion PRO LIVE.

A pesar de la existencia de un gran número de dispositivos y aplicaciones capaces de detectar movimientos y reconocer gestos, ha habido pocos avances en el campo de la inclusión social para personas con diversos tipos de condiciones musculares (limitaciones motoras o discapacidades) que faciliten el acceso a los medios de aprendizaje, información y comunicación. Solo el 10.6% de colombianos con diversas condiciones utilizan medios escritos como revistas, libros y periódicos, mientras que solo el 2.2% de esta población utilizan Internet (Gómez, 2010). Esta última forma de comunicación es una de las más significativas de los

últimos diez años y requiere la interacción física con una computadora.

Mediante el desarrollo de una aplicación informática (software) basada en la visión artificial y el uso de un dispositivo de detección de movimiento basado en modelos 3D, fue posible aumentar la accesibilidad de dispositivos electrónicos como computadoras, palmas y teléfonos móviles para personas con diversas afecciones musculares. Al mismo tiempo, los datos, la información y los conocimientos almacenados en los medios electrónicos o en la Internet son accesibles.

El desarrollo de la aplicación de visión 3D comenzó con la especificación de sus características y requisitos técnicos. Luego, se establecieron los soportes necesarios para el diseño y desarrollo de software para crear la interfaz de usuario que permite a las personas con movilidad limitada utilizar una computadora. El instrumento de visión 3D se incorporó posteriormente a la interfaz de usuario. Finalmente, el desempeño de la aplicación fue evaluado y confirmado en un entorno real.

Es importante reconocer la utilidad de esta herramienta, particularmente a la luz de los desafíos que enfrenta la población objetivo en términos de inclusión social en diversos ámbitos como la educación, el entretenimiento y el empleo. Algunas de las limitaciones identificadas por Jiménez *et al.* (2023) Incluir la capacitación inadecuada de maestros, familias e instituciones para proveer efectivamente servicios educativos y/o laborales. Además, se ha avanzado poco en la elaboración de modelos simplificados de atención para diversas poblaciones. Otros problemas son las prácticas que fomentan activamente el abandono, la segregación y la exclusión en entornos inclusivos, así como la autoexclusión y la marginación. Además, hay una deshumanización de los pacientes, una falta de reconocimiento de los aspectos sociales y emocionales, y la desafección. Los compañeros de esta población enfrentan dificultades debido a limitaciones físicas, edad avanzada y otras enfermedades. Además, a menudo se culpa a las personas y a las familias de sus dificultades, y las instituciones enfrentan altos costos asociados con la provisión de la infraestructura física y tecnológica necesaria para su cuidado. Por último, existe un bajo nivel de apertura entre los centros educativos hacia

su entorno inmediato, entre muchas otras limitaciones.

El presente documento está estructurado de la siguiente manera: en el capítulo 2 se ofrece una visión general de los materiales y procedimientos empleados en el estudio. En el capítulo 3 se resumen las conclusiones y se analizan más a fondo. En el capítulo cuatro se presentan las conclusiones, mientras que en el capítulo cinco se incluyen las referencias bibliográficas.

## 2. MATERIALES Y MÉTODOS

### 2.1 Sensor

Este componente de visión estéreo implica el uso de dos cámaras (una cámara RGB a color y la otra cámara infrarroja IR), así como un proyector infrarrojo para el mapeo 3D del entorno. Su aplicación produce los mapas de profundidad (Zhengyou, 2012) y nubes de puntos necesarios para la reconstrucción del modelo 3D (Castro et al., 2023). Al mismo tiempo, se obtiene información sobre la ubicación de la persona dentro del entorno. A partir de las técnicas de visión artificial, también es posible determinar la velocidad a la que un objeto se mueve y su posición en cada instante en el tiempo (Luo et al., 2023). Además, cuenta con un sistema de comunicación responsable de la transmisión de datos a través de una conexión USB. La figura 1 muestra la distribución de estos componentes dentro del sensor.

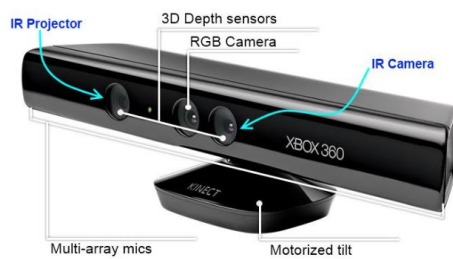


Fig. 1. Kinect sensor (Ramos, 2013).

Se realizan dos tipos de pruebas para este elemento (Miao et al., 2023): (i) para objetos fijos para los que se estudian los límites de tamaño, resolución y calidad. Entre estos objetos físicos, se modelan con precisión rostros humanos de diferentes tipos y a múltiples distancias. Esto se logra en tiempo real, permitiendo una visualización precisa de los gestos realizados. (ii) para mover objetos, el dispositivo recrea un modelo 3D del área donde el usuario se

está moviendo y almacena estos datos para su posterior procesamiento y análisis.

### 2.2 Proceso

Este componente incluye el desarrollo de una interfaz de usuario en el lenguaje de programación C++, que se encarga de extraer las características de las imágenes procesadas y tomar decisiones sobre cómo debe operar el cursor dentro del monitor.

El proceso completo de visión estéreo se compone de seis partes fundamentales (Montalvo, 2010; Mahajam et al., 2023), a saber: (i) adquisición de imágenes; (ii) modelado de cámaras (geometría de aplicación); (iii) extracción de características; (iv) coincidencia de imágenes (características); (v) determinación de distancia (profundidad) y (vi) interpolación.

Los dos primeros son componentes del dispositivo de detección 3D. El primero adquiere la imagen y la transmite al ordenador, mientras que el segundo corresponde a los parámetros intrínsecos de la cámara, como la distancia focal y el modelo de distorsión. Los componentes restantes del proceso de visión estéreo consisten en software que extrae características tales como bordes, áreas y conectividad, entre otros. El sistema luego compara y calcula las diferencias entre las imágenes de ambas cámaras. Después de obtener estos valores y parámetros de la cámara, el mapa de profundidad de cada píxel se determina por triangulación, y si es necesario, la imagen se corrige mediante interpolación (Montalvo, 2010).

El framework Kinect SDK (OpenNI, 2023) fue utilizado para lograr la extracción de las características en la interfaz de usuario mediante los sensores. Este marco proporciona mapas de profundidad, mapas de color, mapas de escena, reconocimiento de gestos y pose de usuario (esqueleto). La Figura 2 muestra una vista de tres capas del concepto OpenNI, que es idéntica a la implementada en el SDK de Kinect, con cada capa representando el siguiente elemento integral: (i) Arriba: representa el software que implementa aplicaciones de interacción natural. (ii) Medio: representa el framework OpenNI y proporciona interfaces de comunicación que interactúan con sensores y componentes de middleware que analizan los datos del sensor. (iii) Abajo: muestra los dispositivos de hardware que capturan los elementos visuales y audibles de la escena.

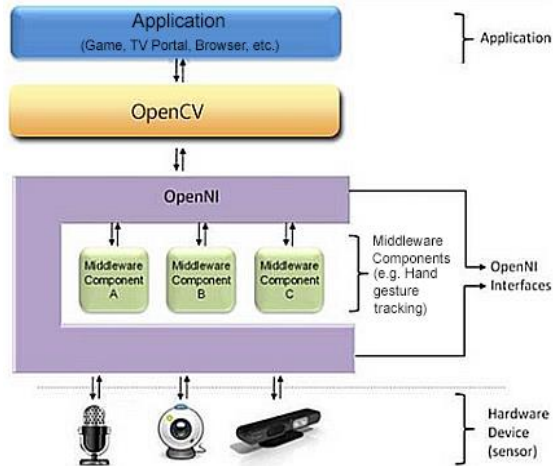


Fig. 2. Capas del OpenNI (OpenNI, 2023).

Una API permite el registro de varios componentes dentro del framework OpenNI. Estos componentes, conocidos como módulos, se utilizan para generar y procesar datos de sensores. Los módulos actualmente soportados incluyen un sensor 3D, cámara RGB, cámara IR y dispositivo de audio (micrófono o matriz de micrófono). El middleware consiste en: (i) análisis de todo el cuerpo: procesa datos sensoriales y genera información relacionada con el cuerpo (típicamente una estructura de datos que caracteriza las articulaciones, la orientación y el centro de masa, entre otros). (ii) Análisis de puntos de mano: procesa datos sensoriales para determinar la ubicación de un punto de mano. (iii) Detección de gestos: reconoce gestos predefinidos (como una mano en movimiento) y notifica a la aplicación. (iv) Analizador de escena: analiza la imagen de escena para producir información como la separación entre el primer plano de la escena (las figuras) y el fondo; coordenadas del plano de tierra y/o identificación individual de las figuras en la escena.

### 3. RESULTADOS Y DISCUSION

#### 3.1 Esquema de diseño

El esquema de diseño tuvo en cuenta tres elementos fundamentales: (i) el primero es el posicionamiento del cursor, que es responsable de obtener la información de coordenadas proporcionada por el SDK y enviarla al sistema de coordenadas C++. (ii) el segundo componente se encarga de reconocer los gestos de brazo del usuario, con los que se ejecutan las acciones asociadas a cada gesto. (iii) Por último, está el diseño de la interfaz de usuario, que cumple con ciertas características que permiten el correcto

funcionamiento e informar al usuario de la acción actual que se realiza.

Para la interacción basada en el movimiento con un ordenador, se utilizó el dispositivo periférico de entrada más frecuente (ratón). El movimiento de la mano del usuario corresponde inmediatamente a la posición del cursor. Si la mano del usuario se mueve hacia la izquierda, el cursor debe viajar hacia el lado izquierdo de la pantalla, y de manera similar para todos los demás movimientos. Basta con mover la mano más cerca y más lejos de la pantalla para hacer clic. Para el clic izquierdo, la mano se acerca a la pantalla y luego vuelve a su posición central; para el clic derecho, el brazo se aleja de la pantalla.

Al detectar movimiento de la mano con el Kinect y el OpenNI SDK, los datos de la mano se presentan como un punto en un sistema de coordenadas de 640 por 480 píxeles. Como se muestra en la Figura 3, el eje X va de -320 a 320 píxeles y el eje Y va de -240 a 240 píxeles.

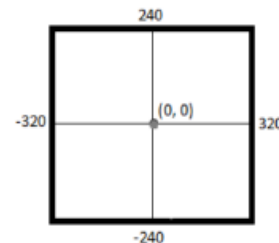


Fig. 3. Coordenadas del sistema utilizadas por el OpenNI SDK.

Para mover el cursor en la pantalla usando datos SDK, los datos deben convertirse a coordenadas. Esto se debe al hecho de que la interfaz C++ solo es capaz de funcionar con un sistema de coordenadas. El origen del sistema de coordenadas se encuentra en la esquina superior izquierda de la pantalla. La Figura 4 es una ilustración del sistema de coordenadas utilizado por C++ para 1920 x 1080.

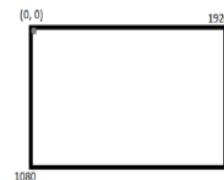


Fig. 4. Coordenadas del sistema utilizadas por C++.

El sistema de coordenadas de C++ para una pantalla de 1920 x 1080 tiene un eje X que va de 0 a 1920 y un eje Y que va de 0 a 1080. Los datos deben convertirse de un rectángulo de 640 x 480 píxeles

con el centro en el centro a un rectángulo de 1920 x 1080 píxeles con el centro en la esquina superior izquierda.

### 3.2 Interfaz del usuario

Para un funcionamiento completo, la aplicación emplea ayudas de accesibilidad similares a MS-Windows, como la lupa y el teclado en pantalla. El usuario debe poder observar el reconocimiento de gestos de la aplicación. Si el usuario intenta hacer clic en un enlace usando el gesto correcto, pero falla, no puede determinar si pasó por alto el enlace o hizo un gesto incorrecto. Para informar al usuario que el gesto se realizó correctamente, la aplicación escribe el gesto reconocido, como "clic", en un cuadro de texto. Como medio de ayudar al usuario a aprender los gestos, la imagen RGB del Kinect se muestra en la interfaz. Al tener acceso a la imagen RGB, los usuarios pueden observar las limitaciones del Kinect y observarse a sí mismos ejecutando los gestos.

La interfaz se divide en las siguientes partes: (i) seguimiento de usuarios para poder recuperar datos. (ii) seguimiento de la mano derecha del usuario, que se utiliza para recuperar las coordenadas de la mano. Esto es necesario para que el cursor pueda moverse en la pantalla en función de la posición de la mano derecha. (iii) reconocimiento gestual, que permite examinar los gestos del brazo derecho.

### 3.3 Resultados de las pruebas de rendimiento de integración de hardware y software

Después de integrar todos los componentes de la aplicación, continuamos con la evaluación y verificación de su funcionamiento en condiciones ambientales reales, como objetos en el campo de visión del usuario y cambios de posición.

Se obtuvieron varios tipos de resultados de estas pruebas. En uno de ellos, se podrían adquirir imágenes de profundidad para reconocer los movimientos y gestos del usuario (Ver Figura 5).

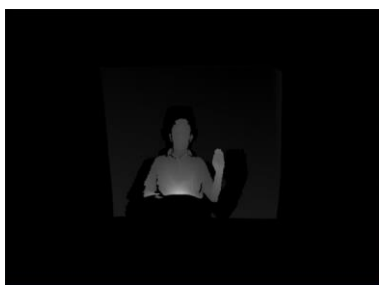


Fig. 5. Imagen de profundidad dada por el sensor Kinect. Fuente: los autores

En las otras pruebas, se recopilaron datos de imágenes de profundidad y color, y se utilizó el método SkeletonTracking del SDK para monitorear las manos con el fin de determinar la posición actual de cada extremidad (Figuras 6 y 7).

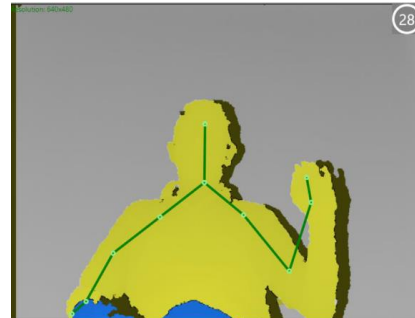


Fig. 6. Imagen de profundidad integrada con SkeletonTracking dada por el sensor Kinect.

Fuente: los autores.

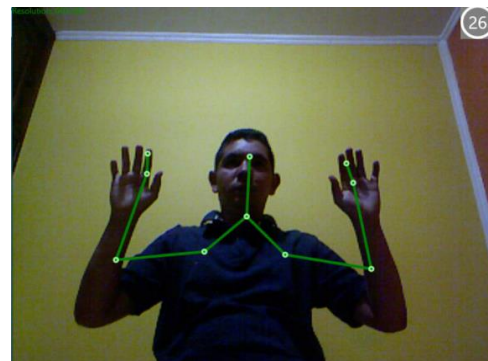


Fig. 7. Imagen a color integrada con el SkeletonTracking dada por el sensor Kinect.

Fuente: los autores.

Para los programas antes mencionados, empleamos las funciones y métodos SDK para el Kinect, con las siguientes funciones utilizadas con mayor frecuencia: `INuiSensor`, `NuiImageResolutionToSize`, and `INuiSensor::NuiImageStreamOpen`. Una vez que obtenemos las coordenadas del punto de mano, todo lo que queda por hacer es escalar las coordenadas de Kinect a las coordenadas de la interfaz del ratón en C++, como se demuestra en las Figuras 5 y 6.

En la interfaz, hay tres indicadores y un botón (Figura 8). Las etiquetas que sirven como indicadores son (i) la etiqueta de estado del sensor, que muestra si el sensor está conectado, desconectado o recién comenzando; (ii) la etiqueta de ID de conexión, que identifica el sensor conectado; y (iii) la etiqueta que muestra las posiciones X e Y de la mano en relación con las coordenadas del sensor.

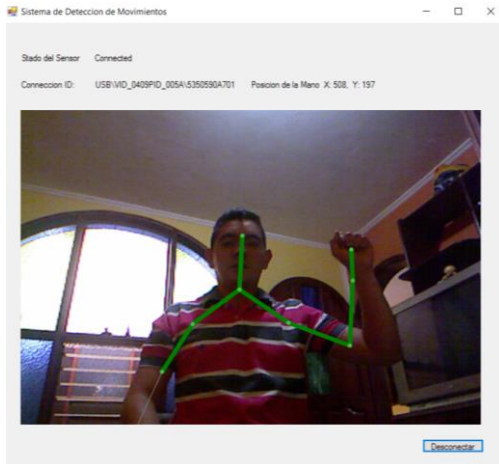


Fig. 8. Interfaz de usuario desarrollada en C++.  
Fuente: los autores.

Se llevaron a cabo dos pruebas para verificar los hallazgos: la primera comparó la rapidez con la que una persona podía presionar el botón izquierdo del ratón y la rapidez con la que podía utilizar el sensor Kinect para realizar un evento de clic. Esta velocidad se midió usando el juego en línea Super Click Tester (Minijuegos, 2023).

La segunda prueba comparó la rapidez y precisión con la que una persona podía anular la selección de un elemento con el ratón y la rapidez con la que podía moverse y hacer clic con el sensor Kinect. Para calcular este resultado, se empleó el juego online Water Balloon War (Juegos Infantiles, 2023). Las figuras 9 y 10 muestran los resultados.

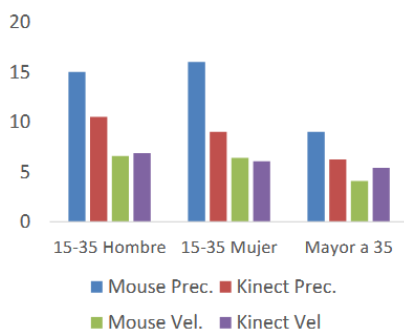


Fig. 9. Resultados de la prueba en porcentaje con respecto a los valores del ratón. Fuente: los autores.

La unidad para la medición de precisión fue el número de golpes. La unidad para la medición de velocidad fue el número de clics por segundo.

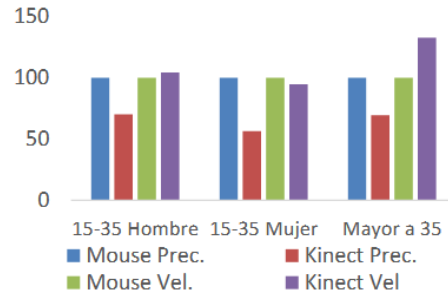


Fig. 10. Los resultados de las pruebas se expresan en porcentaje en comparación con los valores del ratón. Fuente: los autores.

#### 4. CONCLUSIONES

El desarrollo de nuevos productos que puedan satisfacer las necesidades de personas con limitaciones que interfieran con su capacidad de acceder a las computadoras es un factor clave en el proceso de inclusión y aprendizaje.

Los dispositivos de entrada alternativos o auxiliares tienen propiedades operativas muy específicas. Por esta razón, es necesario tener conocimiento previo de los periféricos disponibles, sus características y la tecnología que emplean antes de decidir cuál es el dispositivo más conveniente para una variedad de condiciones musculares.

De acuerdo con la arquitectura del SDK, los resultados de la aplicación tuvieron en cuenta los componentes clave de la aplicación. Para este propósito, la capa de aplicación se dividió en las siguientes secciones: (i) seguimiento de los usuarios con el fin de recuperar sus datos. Se insertó en la vista Kinect a través de un componente de capa OpenNI (UserGenerator). (ii) seguimiento de la mano derecha del usuario para recuperar las coordenadas de posición de ambas manos. Esto es necesario para ajustar el cursor en la pantalla en función de la posición de la mano derecha. HandGenerator, un componente de la capa OpenNI, fue usado. (iii) filtrando los datos de la mano derecha. Elimina las fluctuaciones de fase en los datos utilizados para controlar el cursor. El elemento Filtro, que forma parte de la capa de aplicación, se utilizó para este propósito.

#### REFERENCIAS

Bautista L., Archila J. (2011) *Visión artificial aplicada en sistemas de realidad aumentada*.

- En 3er Congreso Internacional de Ingeniería Mecatrónica – UNAB, Vol. 2, No 1.
- Chen, M., Duan, Z., Lan, Z., & Yi, S. (2023). *Scene reconstruction algorithm for unstructured weak-texture regions based on stereo vision*. Applied Sciences, 13(11), 6407.
- Castro, G. Z., Guerra, R. R., & Guimarães, F. G. (2023). *Automatic translation of sign language with multi-stream 3D CNN and generation of artificial depth maps*. Expert Systems with Applications, 215, 119394.
- Figuroa, Y., Arias, L., Mendoza, D., Velasco, N., Rea, S., & Hallo, V. (2018). *Autonomous video surveillance application using artificial vision to track people in restricted areas*. In Developments and Advances in Defense and Security: Proceedings of the Multidisciplinary International Conference of Research Applied to Defense and Security (MICRADS 2018) (pp. 58-68). Springer International Publishing.
- Gómez, J. (2010) *Discapacidad en Colombia: reto para la inclusión en capital humano*. Colombia Líder. Bogotá: Fundación Saldarriaga Concha.
- Huynh-The, T., Pham, Q. V., Pham, X. Q., Nguyen, T. T., Han, Z., & Kim, D. S. (2023). *Artificial intelligence for the metaverse: A survey*. Engineering Applications of Artificial Intelligence, 117, 105581.
- Jones, C. R., Trott, S., & Bergen, B. (2023). *EPITOME: Experimental Protocol Inventory for Theory Of Mind Evaluation*. In First Workshop on Theory of Mind in Communicating Agents.
- Juegos infantiles (2023). *Juegos infantiles en línea*. [Online]. Disponible en: <http://www.juegosinfantiles.com/locos/guerra deglobosdeagua.html> Fecha de acceso: septiembre de 2023.
- Luo, X., Sun, Q., Yang, T., He, K., & Tang, X. (2023). *Nondestructive determination of common indicators of beef for freshness assessment using airflow-three dimensional (3D) machine vision technique and machine learning*. Journal of Food Engineering, 340, 111305.
- Mahajan, H. B., Uke, N., Pise, P., Shahade, M., Dixit, V. G., Bhavsar, S., & Deshpande, S. D. (2023). *Automatic robot Manoeuvres detection using computer vision and deep learning techniques: A perspective of internet of robotics things (IoRT)*. Multimedia Tools and Applications, 82(15), 23251-23276.
- Mauri, C. (2004). *Interacción persona-ordenador mediante cámaras Webcam*. In J. Lorés and R. Navarro (Eds.), Proceedings of V Congress Interacción Human-Computer, pp. 366–367. Lleida, Spain: Arts Gràfiques Bobalà SL.
- Miao, R., Liu, W., Chen, M., Gong, Z., Xu, W., Hu, C., & Zhou, S. (2023). *Occdepth: A depth-aware method for 3d semantic scene completion*. arXiv preprint arXiv:2302.13540.
- Minijuegos (2023). *Minijuegos en línea*. [Online]. Disponible en: <http://www.minijuegos.com/juego/super> Fecha de acceso: septiembre de 2023.
- Montalvo, M. (2010). *Técnicas de visión estereoscópica para determinar la estructura tridimensional de la escena*. Doctoral tesis, Universidad Complutense de Madrid, España.
- OpenNI (2023). *OpenNI user guide*. [Online]. Disponible en: [https://github.com/OpenNI/OpenNI/blob/master/Documentation/OpenNI\\_UserGuide.pdf](https://github.com/OpenNI/OpenNI/blob/master/Documentation/OpenNI_UserGuide.pdf) Fecha de acceso: septiembre de 2023.
- Ramos, D. (2013). *Estudio cinemático del cuerpo humano mediante Kinect*. Trabajo de grado en Telecomunicaciones, Escuela Técnica De Telecomunicaciones, Universidad Politécnica de Madrid, España.
- Sánchez, J., Cardona, H. & Jiménez, J. (2023) *Potencialidades del uso de las herramientas informáticas para la optimización del acceso a la oferta educativa de personas adultas con trastornos neuromusculares que habitan en el Área Metropolitana del Valle de Aburrá*. En: XX Congreso Latino - Iberoamericano de Gestión Tecnológica y de la Innovación - ALTEC 2023, Paraná, Argentina.
- Song, Y., Demirdjian, D. (2010) *Continuous body and hand gesture recognition for natural human-computer interaction*. ACM Transactions on Interactive Intelligent Systems, vol. 1, No. 1, pp. 111-148.
- Villaverde, I. (2009). *On computational intelligence tools for vision based navigation of mobile robots*. Doctoral thesis. , Department of Computer Science and Artificial Intelligence, University of the Basque Country, España.
- Zhengyou, Z. (2012). *Microsoft kinect sensor and its effect*. Multimedia at Work, IEEE Computer Society, pp. 4-10.