3D ARTIFICIAL VISION TECHNOLOGY FOR DETECTING MOVEMENTS IN PEOPLE WITH MUSCULAR DISABILITIES THROUGH A COMPUTER APPLICATION

TECNOLOGÍA DE VISIÓN ARTIFICIAL 3D PARA DETECTAR MOVIMIENTOS EN PERSONAS CON CONDICIONES MUSCULARES DIVERSAS A TRAVÉS DE UN APLICATIVO INFORMÁTICO

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

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 ingresarse 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 muscular conditions, educational inclusion, digital transformation, artificial intelligence.

1. INTRODUCTION

Currently, artificial vision is used in a wide range of computer applications that are primarily industry and research-oriented. These applications include, among others, autonomous navigation (Villaverde, 2009), surveillance systems (Figueroa *et al.,* 2018), augmented reality (Bautista & Archila, 2011), and social inclusion (Mauri, 2004).

According to Huynh *et al*. (2023), devices based on artificial vision permit interaction with the computer via various techniques; two of them are: (i) Eye movement: in this category, there are several developments, including the BET2.0, which allows the user to control the computer cursor by means of a device based on artificial vision and software installed on a personal computer. This application primarily benefits those with severe mobility limitations who are unable to use conventional peripherals such as a mouse and keyboard (2023, Jones *et al*.). (ii) Gesture and hand movement detection: this category comprises both body movement recognition and hand gesture recognition. This type of software is based on motion detection through stereo vision, which can be categorized into two groups (Chen *et al*., 2023): detection based on models of the hand and arm, and detection of hand gestures (finger positions, orientation, etc.) (Song & Demirdjian, 2010). For this type of technique, there are several devices on the market, including the Kinect, iSense™3D scanner, and the Xtion PRO LIVE.

Despite the existence of a large number of devices and applications capable of detecting movement and recognizing gestures, there have been few developments in the field of social inclusion for people with various kinds of muscular conditions (motor limitations or disabilities) that facilitate easy access to learning, information, and communication media. Only 10.6% of Colombians with diverse conditions use written media such as magazines, books, and newspapers, whilst only 2.2% of this population use the Internet (Gómez, 2010). This last form of communication is one of the most significant of the past ten years and requires physical interaction with a computer.

Through the development of a computer application (software) based on artificial vision and the use of a motion detection device based on 3D models, it was possible to increase the accessibility of electronic devices such as computers, palms, and mobile phones for people with various muscular conditions. At the same time, data, information, and knowledge stored on electronic media or the Internet were accessible.

The development of the 3D vision application began with the specification of its characteristics and technical requirements. Then, the necessary supports were established for the design and development of software to create the user interface that enables individuals with limited mobility to use a computer. The 3D vision instrument was subsequently incorporated into the user interface. Finally, the application's performance was evaluated and confirmed in a real-world setting.

It is important to acknowledge the usefulness of this tool, particularly in light of the challenges faced by the target population in terms of social inclusion within various domains such as education, entertainment, and employment. Some of the limitations identified by Jiménez *et al.* (2023) include the inadequate training of teachers, families, and institutions to effectively provide educational and/or labor services. Additionally, there has been limited progress in developing simplified models of care for diverse populations. Other challenges include practices that actively encourage abandonment, segregation, and exclusion within inclusive settings, as well as self-exclusion and marginalization. Furthermore, there is a dehumanization of patients, a lack of recognition of social and emotional aspects, and disaffection. Companions of this population face difficulties due to physical limitations, advanced age, and other illnesses. In addition, individuals and families are often blamed for their difficulties, and institutions face high costs associated with providing the necessary physical and technological infrastructure for their care. Lastly, there is a low level of openness among educational centers towards their immediate environment, among many other limitations.

This paper is structured as follows: chapter two provides an overview of the materials and procedures employed in the study. Chapter three summarizes the findings and further analysis. Chapter four provides the findings and conclusions,

while chapter five encompasses the bibliographic references.

2. MATERIALS AND METHODS

2.1 Sensing

This stereo vision component involves the use of two cameras (one color-RGB camera and the other infrared-IR camera) as well as an infrared projector for 3D mapping of the environment. Their application yields the depth maps (Zhengyou, 2012) and point clouds required for the reconstruction of the 3D model (Castro *et al*., 2023). Concurrently, information about the person's location within the environment is obtained. From artificial vision techniques, it is also possible to determine the speed at which an object moves and its position at each instant in time (Luo *et al.,* 2023). In addition, it has a communication system responsible for data transmission over a USB connection. Figure 1 depicts the distribution of these components within the sensor.
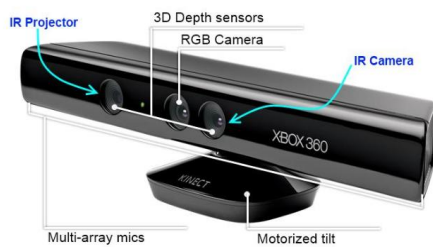

Fig. 1. Sensor Kinect (Ramos, 2013).

Two kinds of tests are conducted for this element (Miao *et al*., 2023): (i) for fixed objects for which the size, resolution, and quality limits are studied. Among these physical objects, human faces of different types and at multiple distances are accurately modeled. This is accomplished in real time, permitting accurate visualization of performed gestures. (ii) for moving objects, the device recreates a 3D model of the area where the user is moving and stores this data for subsequent processing and analysis.

2.2 Processing

This component includes the development of a user interface in the C++ programming language, which is responsible for extracting the characteristics of the processed images and making decisions regarding how the cursor should operate within the monitor.

The complete stereo vision process is composed of six fundamental parts (Montalvo, 2010; Mahajam *et al*., 2023), namely: (i) image acquisition; (ii) camera modeling (application geometry); (iii) feature extraction; (iv) image matching (features); (v) distance determination (depth) and (vi) interpolation.

The first two are components of the 3D sensing device. The first acquires the image and transmits it to the computer, while the second corresponds to the intrinsic camera parameters, such as the focal length and distortion model. The remaining components of the stereo vision process consist of software that extracts features such as edges, areas, and connectivity, among others. The system then compares and calculates the differences between the images from both cameras. After obtaining these values and camera parameters, the depth map of each pixel is determined by triangulation, and if necessary, the image is corrected through interpolation (Montalvo, 2010).

The Kinect SDK framework (OpenNI, 2023) was used to achieve the extraction of the features in the user interface by means of the sensors. This framework provides depth maps, color maps, scene maps, gesture recognition and user pose (skeleton). Figure 2 depicts a three-layer view of the OpenNI concept, which is identical to that implemented in the Kinect SDK, with each layer representing the following integral element: (i) Top: represents the software that implements natural interaction applications. (ii) Middle: represents the OpenNI framework and provides communication interfaces that interact with sensors and *middleware* components that analyze sensor data. (iii) Bottom: displays the hardware devices that capture the scene's visual and audible elements.
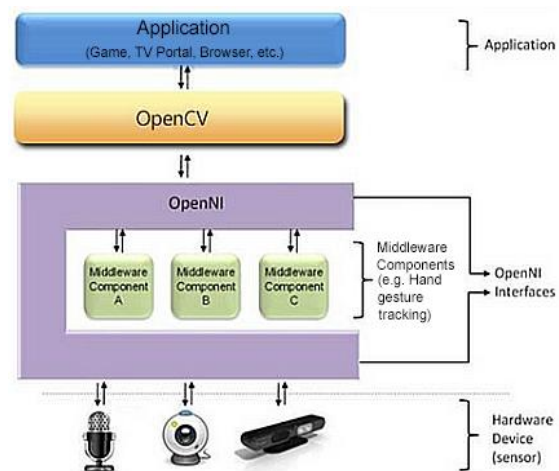

Fig. 2. OpenNI Layers (OpenNI, 2023).

An API permits the registration of multiple components within the OpenNI framework. These components, referred to as modules, are used to generate and process sensor data. Currently supported modules include a 3D sensor, RGB camera, IR camera, and audio device (microphone or microphone array). The middleware consists of: (i) whole body analysis: processes sensory data and generates body-related information (typically a data structure characterizing the joints, orientation, and center of mass, among others). (ii) Hand point analysis: processes sensory data to determine the location of a hand point. (iii) Gesture detection: recognizes predefined gestures (such as a moving hand) and notifies the application. (iv) Scene analyzer: analyzes the scene image to produce information such as separation between the foreground of the scene (the figures) and the background; coordinates of the ground plane and/or individual identification of the figures in the scene.

3. RESULTS AND DISCUSSION

3.1 Design scheme

The design scheme took into account three fundamental elements: (i) the first is cursor positioning, which is responsible for obtaining the SDK-provided coordinate information and forwarding it to the C++ coordinate system. (ii) the second component is responsible for recognizing the arm gestures of the user, with which the actions associated with each gesture are executed. (iii) Lastly, there is the design of the user interface, which meets certain characteristics that enable correct operation and inform the user of the current action being performed.

For motion-based interaction with a computer, the most frequent input peripheral device (mouse) was used. The movement of the user's hand corresponds immediately to the position of the cursor. If the user's hand is moved to the left, the cursor should travel to the left side of the display, and similarly for all other movements. Simply move the hand closer and farther away from the screen to click. For the left click, the hand is brought closer to the screen and then returned to its central position; for the right click, the arm is moved away from the screen.

When detecting hand motion with the Kinect and the OpenNI SDK, the hand data is presented as a point in a 640-by-480-pixel coordinate system. As shown in Figure 3, the X-axis ranges from -320 to 320

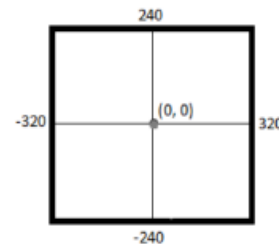pixels and the Y-axis ranges from -240 to 240 pixels.

Fig. 3. System coordinates used by the OpenNI SDK.

To move the cursor on the screen using SDK data, the data must be converted to coordinates. This is due to the fact that the C++ interface is only capable of functioning with a coordinate system. The coordinate system's origin is located in the upper left corner of the display. Figure 4 is an illustration of the coordinate system used by C++ for 1920 x 1080.
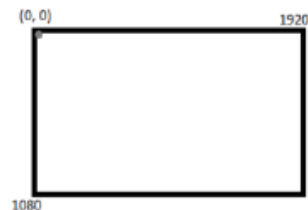

Fig. 4. System coordinates used by C++.

C++'s coordinate system for a 1920 x 1080 display has an X-axis ranging from 0 to 1920 and a Y-axis ranging from 0 to 1080. The data must be converted from a 640 x 480-pixel rectangle with the center in the middle to a 1920 x 1080-pixel rectangle with the center in the upper left corner.

3.2 User interface

For comprehensive operation, the application employs MS-Windows-like accessibility aids such as the magnifying glass and the on-screen keyboard. The user should be able to observe the application's gesture recognition. If the user attempts to click on a link using the correct gesture, but fails, they cannot determine whether they overlooked the link or made an incorrect gesture. In order to inform the user that the gesture was conducted correctly, the application writes the recognized gesture, such as "click", in a text box. As a means of assisting the user in learning the gestures, the RGB image of the Kinect is displayed on the interface. By having access to the RGB image, users can observe the limitations of the Kinect and observe themselves executing the gestures.

The interface is divided into the following parts: (i) user tracking to be able to retrieve data. (ii) tracking of the user's right hand, which is used to retrieve the hand's coordinates. This is required so that the cursor can be moved on the screen based on the position of the right hand. (iii) gesture recognition, which permits the examination of right-arm gestures.

3.3 Hardware & software integration performance test results

After integrating all of the application's components, we continued with the evaluation and verification of its operation under actual environmental conditions, such as objects in the user's field of vision and position changes.

Several kinds of results were obtained from these tests. In one, depth images could be acquired in order to recognize the user's movements and gestures (See Figure 5).
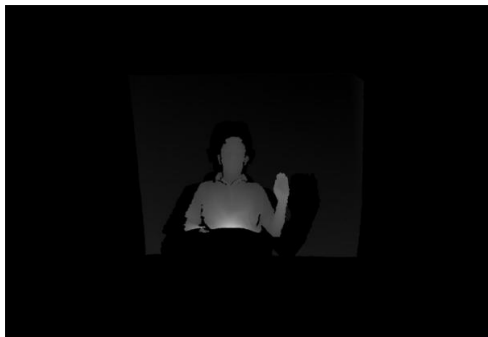

*Fig. 5. Depth image given by the Kinect sensor.
Source: the authors*

In the other tests, depth and color images data was collected, and the SDK's *SkeletonTracking* method was used to monitor the hands in order to determine the current position of each limb (Figures 6 and 7).
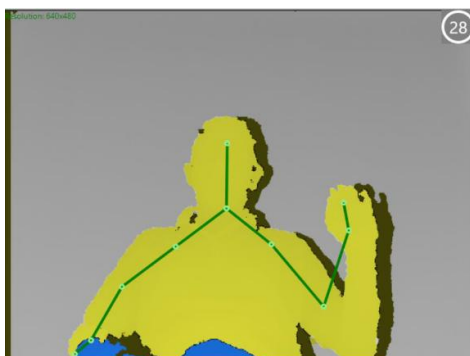

*Fig. 6. Depth image integrated with
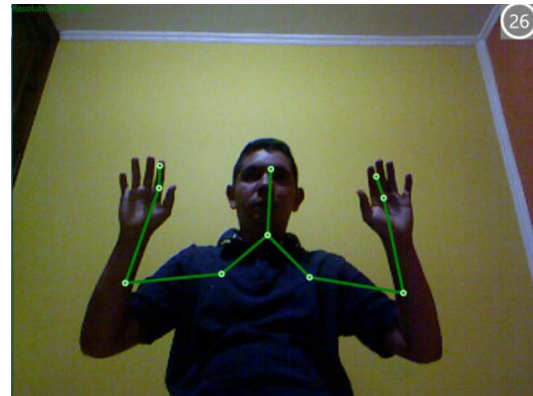SkeletonTraking given by the Kinect sensor.
Source: the authors.*

*Fig. 7. Imagen a color integrada con el
SkeletonTraking dada por el sensor Kinect.
Fuente: los autores.*

For the aforementioned programs, we employed the SDK functions and methods for the Kinect, with the following functions used most frequently: *INuiSensor, NuiImageResolutionToSize,* and *INuiSensor::NuiImageStreamOpen.*
Once we obtain the hand point coordinates, all that is left to do is scale the Kinect coordinates to the mouse interface coordinates in C++, as demonstrated in Figures 5 and 6.

On the interface, there are three indicators and a button (Figure 8). The labels that serve as indicators are (i) the sensor status label, which shows whether the sensor is connected, disconnected, or just starting; (ii) the connection ID label, which identifies the connected sensor; and (iii) the label that shows the hand's X and Y positions in relation to the sensor coordinates.
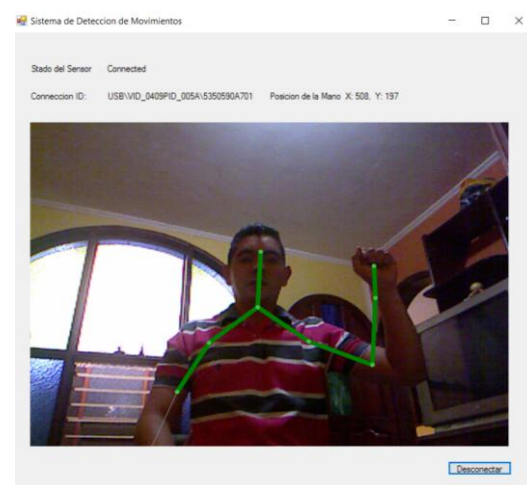

*Fig. 8. User interface developed in C++. Source:
the authors.*

Two tests were conducted to verify the findings: the first compared how quickly a person could press the

left mouse button to how quickly they could utilize the Kinect sensor to perform a click event. This speed was measured using the online game Super Click Tester (Minijuegos, 2023).

The second test compared how quickly and accurately a person could deselect an item using the mouse to how quickly they could move and click using the Kinect sensor. To calculate this outcome, the online game Water Balloon War (Juegos Infantiles, 2023) was employed. Figures 9 and 10 display the results.
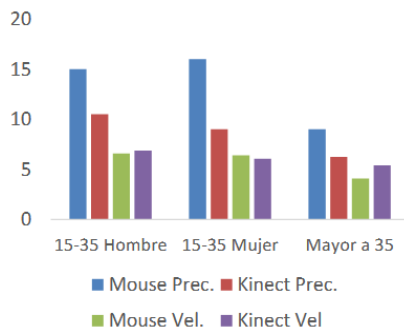

Fig. 9. Test results in percentage with respect to the mouse values. Source: the authors.

The unit for the accuracy measurement was the number of hits. The unit for the speed measurement was the number of clicks per second.
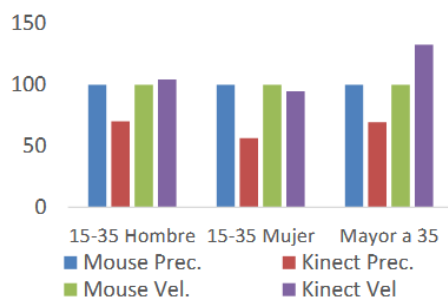

Fig. 10. Test results expressed as a percentage compared to mouse values. Source: the authors

4. CONCLUSIONS

The development of new products that can meet the needs of people with limitations that interfere with their ability to access computers is a key factor in the inclusion and learning process.

Alternative or auxiliary input devices have very specific operational properties. For this reason, it is necessary to have prior knowledge of the available peripherals, their characteristics, and the technology they employ prior to deciding which is the most

convenient device for a variety of muscular conditions.

According to the SDK's architecture, the application's results took into consideration the application's key components. For this purpose, the application layer was divided into the following sections: (i) tracking users in order to retrieve their data. It was inserted in the Kinect View via an OpenNI layer component (*UserGenerator*). (ii) tracking of the user's right hand to retrieve the positional coordinates of both hands. This is required in order to adjust the cursor on the screen based on the position of the right hand. *HandGenerator*, a component of the OpenNI layer, was used. (iii) filtering the data of the right hand. It eliminates phase fluctuations in the data used to control the cursor. The element *Filter*, which is part of the application layer, was utilized for this purpose.

REFERENCES

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.

Figueroa, 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*. InJ. 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 linea*. [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 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.