

CONSTRUCCIÓN DE UN ESCÁNER TRIDIMENSIONAL USANDO C++ Y HARDWARE LIBRE^a

CONSTRUCTION OF A THREE-DIMENSIONAL SCANNER USING C ++ AND FREE HARDWARE

CRISTIAN GAMBOA CAICEDO^b, CARLOS W. SÁNCHEZ^b

Recibido 31-10-2016, aceptado 07-02-2017, versión final 21-02-2017.

Artículo Investigación

RESUMEN: En este trabajo, se desarrolló un sistema económico de escaneo tridimensional (3D) que permite obtener información de la estructura física de un determinado objeto a partir de la cual, se genera un modelo digital. Este sistema opera proyectando un haz de luz láser en forma de línea sobre el objeto, mientras rota sobre su propio eje (eje z), el haz de luz crea un perfil del objeto, el cual es capturado mediante video de alta definición (HD) por una cámara. A través del procesamiento digital de las imágenes capturadas se obtienen los vértices del objeto y se construye el modelo tridimensional. Para llevar a cabo esta tarea, el escáner 3D está formado por un conjunto de elementos de hardware y software libre como lo son: la plataforma Arduino, que permite realizar el control del hardware de escaneo, y un software desarrollado en C++ que implementando las librerías OpenCv y OpenGL, permite la realización de las tareas de procesamiento y análisis de imágenes. Estos elementos hacen que este desarrollo se constituya como incursión económica en la tecnología de escaneo láser tridimensional.

PALABRAS CLAVE: escáner; láser; modelo tridimensional; hardware libre.

ABSTRACT: This paper shows the development of an economic three-dimensional scan system (3D) that provides information about the physical structure of a particular object from which a digital model is generated. This system operates by projecting a laser beam as a line on the object as it rotates on its own axis (z-axis), the beam creates a profile of the object, which is captured by high definition (HD) videocamera. By means of digital processing of the captured images, the object vertices are obtained and the three-dimensional model is constructed. To accomplish this task, the 3D scanner consists of a set of open source hardware and software, such as: the Arduino platform which allows to control of scanning and a software developed in C++ that implementing bookstores OpenCV and OpenGL, allows the tasks of processing and image analysis. These elements make this development, an economic foray into three-dimensional laser scanning technology.

KEYWORDS: acanner; laser; three-dimensional model; free hardware.

^aGamboa, C. E. & Sánchez, C. W. (2017). Construcción de un escáner tridimensional usando C++ y hardware libre. *Revista de la Facultad de Ciencias*, 6 (1), 9–23. DOI: 10.15446/rev.fac.cienc.v6n1.60812

^bGrupo de investigación KIMSA, Facultad de Ingeniería, Fundación Universitaria Católica Lumen Gentium-UNICATOLICA, Cra. 122 No. 12- 459, Cali, Colombia. cegamboa@unicatolica.edu.co

1. INTRODUCCIÓN

Es muy común encontrarse con modelos digitales, que pueden ser creaciones provenientes de la imaginación, y en ocasiones, representaciones de objetos de la vida real, que requieren ser estudiados preservando una de sus características más importantes, la integridad. Éste es un tema que siempre ha causado gran interés entre diferentes campos como: la arqueología, la medicina, la ingeniería, entre otros, debido a la posibilidad de observar y manipular objetos a detalle, sin tener contacto con los mismos.

El desarrollo de modelos tridimensionales presenta dificultades técnicas cuando se trata de modelos orgánicos con muchas curvas, como es el caso de animales o rostros humanos, aún más cuando lo que se requiere es recrear estos detalles en el modelo partiendo de un objeto conocido y manteniendo sus dimensiones, colores y/o texturas. Para esto surgen soluciones como los escáneres 3D, que permiten capturar las dimensiones reales del objeto punto a punto de manera que se pueda generar una imagen tridimensional del mismo, es decir, una representación virtual de un objeto real.

En este artículo, se dará a conocer un sistema de escaneo láser tridimensional (3D) multipropósito, construido a través del lenguaje C++, las librerías openCv y OpenGL, y la tarjeta de circuito programable Arduino, herramientas de software y hardware libre, respectivamente. Este dispositivo utiliza procesos matemáticos, como el manejo de matrices y el cambio de coordenadas, para obtener información de un objeto real, y luego generar un modelo tridimensional constituido por polígonos, el cual puede ser estudiado en diferentes entornos de modelado.

Este artículo está dividido por secciones, con el fin de contemplar todos los aspectos implicados en el desarrollo del producto. La sección 2, describe los conceptos utilizados como base para el desarrollo del proyecto. En la sección 3, se mencionan los componentes utilizados para la elaboración del sistema de escaneo. En la sección 4, se muestran los procesos necesarios para la generación de los modelos tridimensionales, y por último, en la sección 5, se muestran algunos de los resultados obtenidos.

2. CONCEPTOS PRINCIPALES

En esta sección, se describen los conceptos principales, utilizados para el desarrollo del sistema de escaneo descrito en este artículo.

2.1. Modelado 3D

El modelado tridimensional es el proceso que permite crear estructuras digitales en tres dimensiones, mediante el uso de computadores y empleando software especializado para este propósito (Marrero,

2009). Existen diferentes técnicas de modelado, las más importantes se encuentran: la reconstrucción por superficies, en donde la estructura es construida a partir de superficies generadas por polinomios, y la reconstrucción por polígonos, que básicamente permite la construcción de modelos a partir de triángulos (Marrero, 2009).

2.1.1. Modelado de curvas y superficie

El modelo de curvas y superficies permite la construcción de los modelos a través de puntos, curvas, líneas, y superficies (Marrero, 2009; Galdames, 2011; Salgado *et al.*, 2008).

2.1.2. Modelado de polígonos

Un polígono es un conjunto de líneas unidas que no se cruzan y son unidas por puntos que encierran un área simple. La mayoría de los modelos tridimensionales están compuestos por arreglos triangulares conocido como mallas triangulares (Marrero, 2009). Esta técnica permite obtener información del modelo como: el volumen, la transparencia, la masa, otros, (Baldassari, 2013; Scala, 2012).

2.2. Escáner 3D

Un escáner 3D es un dispositivo hardware de captura, que con ayuda de un software de procesamiento de imágenes, transfiere la información de un objeto real, en forma de nube de puntos, a un espacio virtual tridimensional (Marrero, 2009).

Generalmente, los escáneres tridimensionales se clasifican en dos tipos: escáner 3D de contacto y escáner 3D sin contacto. El primero, captura la información por medio de una sonda que actúa en puntos de contacto sobre el objeto, traduciendo estos puntos en coordenadas del objeto virtual. Esta técnica tiene como desventaja la posibilidad de una alteración en la integridad del objeto. El segundo, no requiere tocar el objeto para completar su tarea, pero presenta dificultades en el análisis de detalles como hoyos y oclusiones (Marrero, 2009). Este último tipo de escáner, se clasifica en dos tipos: activos y pasivos. Una breve descripción de estos se desarrollará a continuación.

2.2.1. Escáner pasivo

El escáner pasivo utiliza la intensidad de luz para obtener la profundidad y generar la reconstrucción del modelo (Marrero, 2009).

2.2.2. Escáner activo

El escáner activo irradia algún tipo energía sobre el objeto, para detectar la posición de cada uno de los puntos del mismo, y obtener el modelo deseado (Marrero, 2009).

El escáner activo recibe su nombre dependiendo de la tecnología que se implemente, así se presenta sus diferentes tipos:

- Escáner 3D por luz estructurada: consiste en la proyección de un patrón de luz sobre un objeto, y el registro de información por parte de una cámara.
- Escáner 3D láser: consiste en la proyección de un haz de luz sobre el objeto, que es reflejado por el mismo, y luego esta información es tomada a partir de una relación distancia/tiempo que depende del rebote de la luz.
- Escáner 3D por triangulación: consiste en la proyección de un haz de luz láser sobre el objeto y el uso de una cámara para el registro de información. La ubicación de la cámara, el láser, y el objeto que generalmente reposa sobre una superficie giratoria, forman un triángulo, lo que da su nombre a esta técnica. Éste será el tipo de escáner implementado en este artículo.
- Escáner 3D de sonar: consiste en la emisión de una onda electromagnética, que rebota sobre el objeto, y permite obtener la ubicación de los puntos del mismo.

Éste es el tipo de escáner que se implementó para el desarrollo de la herramienta aquí descrita. Por esto, se requiere que el objeto a escanear presente un alto nivel de opacidad para un adecuado proceso de captura.

3. COMPONENTES DEL SISTEMA

En esta sección, se hace una descripción del diseño, en cuanto a software y hardware, del sistema descrito.

3.1. Software

El componente de software se describe a través de diagramas de flujo, como se muestra a continuación:

- El diagrama 1 (ver Figura 1) describe el componente que se encarga del procesamiento de las imágenes capturadas y construcción del modelo tridimensional.
- El diagrama 2 (ver Figura 2) describe el componente que se encarga del control de hardware de escaneo, permitiendo la comunicación con el software.

Para la elaboración de este componente de software, se implementaron las librerías de procesamiento de imágenes y generación de escenas tridimensionales, OpenCV (*Open Source Computer Vision*) y OpenGL (*Open Graphics Library*), respectivamente. La primera es una librería de código abierto utilizada para la implementación de las técnicas de visión por computadora, mientras que la

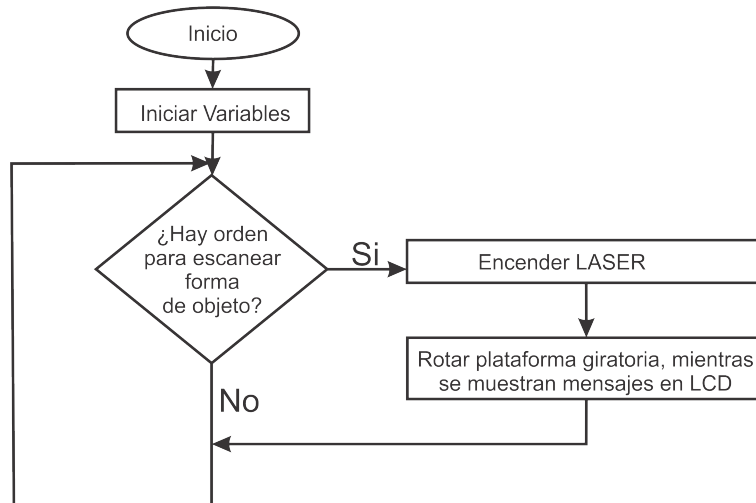


Figura 1: Diagrama de flujo 1, control del hardware de escaneo. Fuente: Elaboración propia.

segunda, es una librería gráfica que permite generar escenas en dos y tres dimensiones (Bradski & Kaebler, 2008).

3.2. Hardware

Los elementos hardware del equipo desarrollado para la generación de modelos tridimensionales, será agrupado en elementos de rotación, elementos de captura y elementos de interacción con el usuario. Todo esto, con el fin de tener una mayor claridad del funcionamiento del sistema.

3.2.1. Elementos de rotación

Los elementos utilizados para la rotación del objeto a escanear, se describen a continuación:

- **Arduino:** es una plataforma de desarrollo computacional de código abierto, que se constituye de componentes como: micro-controlador, entradas y salidas de tipos analógicas y digitales, Memoria RAM (*Random Access Memory*), entre otros (Arduino, 2016). Debido a su facilidad de uso y la gran cantidad de librerías, esta plataforma es muy utilizada en proyectos de robótica, domótica, entre otros (Crespo & Fernández, 2014). El modelo de Arduino utilizado en este artículo es el Arduino Mega 2560 (Monk, 2012). Esta tarjeta permite el control del hardware a partir del software.
- **Motor Shield:** es un periférico electrónico, compatible con la plataforma Arduino, que permite el manejo de relés, solenoides y motores paso a paso (Monk, 2012). En este proyecto, el dispositivo *MotorShield* junto con la plataforma Arduino, permiten el movimiento de la plataforma giratoria.

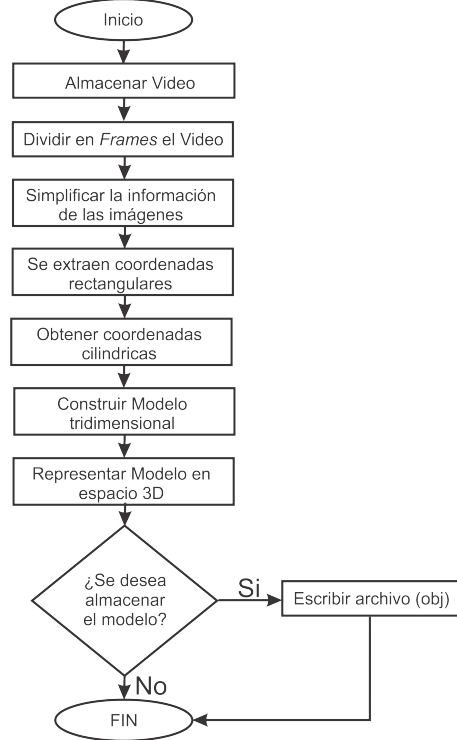


Figura 2: Diagrama de flujo 2, componente de software. Fuente: Elaboración propia.

- Este periférico, está basado en el chip L298, el cual es un integrado formado por un doble puente H, lo que le permite el manejo de dispositivos como: relés, solenoides, motores DC y motores paso a paso (Monk, 2012). Un puente H, es un dispositivo de tipo electrónico compuesto principalmente por transistores, que permite el control del sentido de giro de un motor DC (Carletti, 2007; Conti, 2005).
- Plataforma giratoria: La plataforma giratoria utilizada para generar la rotación de 360 grados del objeto, se compone de un disco de acrílico acoplada mecánicamente a un motor paso a paso que lo impulsa. Un motor paso a paso es un tipo de motor que avanza dando pasos discretos en ángulos dados por las especificaciones del motor, permitiéndose dar giros completos o fracciones de vuelta (Palazzesi, 2008).

3.2.2. Elementos de captura

Los elementos que permiten la captura de imágenes del objeto a escanear, se describen a continuación:

- Cámara: la cámara utilizada para la obtención de las imágenes es una Logitech C525, la cual fue seleccionada por su gran desempeño presentado en proyectos relacionados con el tema de escaneo de imágenes (Logitech, 2010; Logitech, 2016).

- Láser: el haz de luz láser que se proyecta sobre el objeto, es generado por un apuntador rojo con longitud de onda de $650\text{ nm} \pm 10$ potencia máxima de 5mW . Debido a que este elemento ofrece un punto, se presenta la necesidad de generar el fenómeno de refracción a través un prisma cilíndrico, que es un elemento con capacidades para reflejar, dispersar y polarizar la luz, y permite obtener un haz sobre el objeto a escanear (Marrero, 2009). En la Figura 3, se muestra un esquema del arreglo láser.

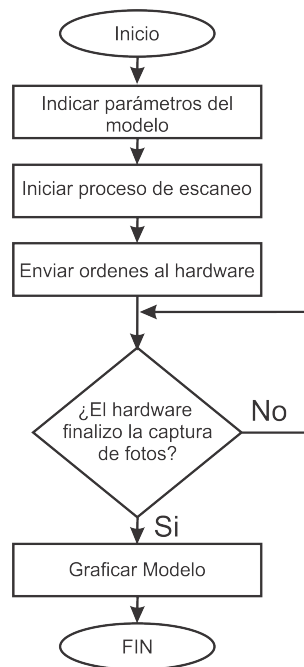


Figura 3: Esquema de arreglo laser. Fuente: Elaboración propia.

3.2.3. Elementos de interacción con el usuario

El elemento de interacción con el usuario implementado dentro del usuario corresponde a una pequeña pantalla LCD (*liquid crystal display*), la cual permite mostrar el estado de funcionamiento del escáner.

3.3. Esquema eléctrico del hardware

En la Figura 4 se muestra el esquema eléctrico del hardware del sistema de escaneo descrito en este artículo.

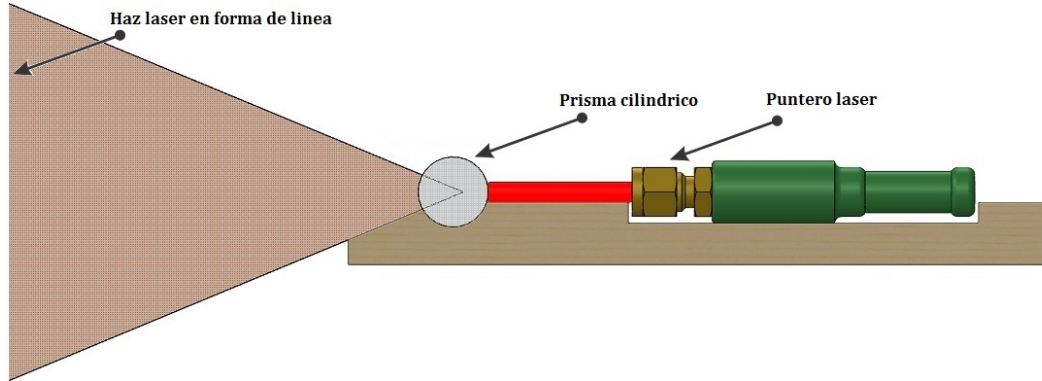


Figura 4: Esquema eléctrico del hardware del sistema. Fuente: Elaboración propia.

3.4. Comunicación entre software y hardware

En el mundo de las redes, existen dos tipos de transmisión de datos: comunicación serial y comunicación paralela. La primera, consiste en la transmisión de datos a través de una sola línea, por lo que sólo puede enviarse un bit a la vez. La segunda, transmite simultáneamente la información a través de las ocho líneas, por lo que puede enviar un byte a la vez (Llamas, 2014).

En el desarrollo de este proyecto, se utilizó la comunicación serial, con el fin de enviar información desde el programa de digitalización hasta la plataforma Arduino que controla el hardware, y viceversa.

Esta comunicación, permitió el planteamiento de la lógica del sistema como se muestra en la Figura 3.

4. CONSTRUCCIÓN DEL MODELO TRIDIMENSIONAL

El escáner desarrollado consta de una fuente de luz láser y una cámara digital de alta definición ubicados sobre el mismo plano y desfasados un ángulo θ uno con respecto al otro, tal y como se muestra en la Figura 5.

En el proceso de captura de información a través de imágenes, se implementa el principio de triangulación, el cual permite aprovechar la rotación de la plataforma tomando los datos de cada uno de los perfiles de manera que se pueda realizar la reconstrucción completa del objeto. El haz de luz debe ser enfocado de tal forma que tenga alineación con el eje Y de la plataforma, además, la figura debe ser ubicada en el centro de la superficie giratoria para determinar con mayor precisión las dimensiones del mismo.

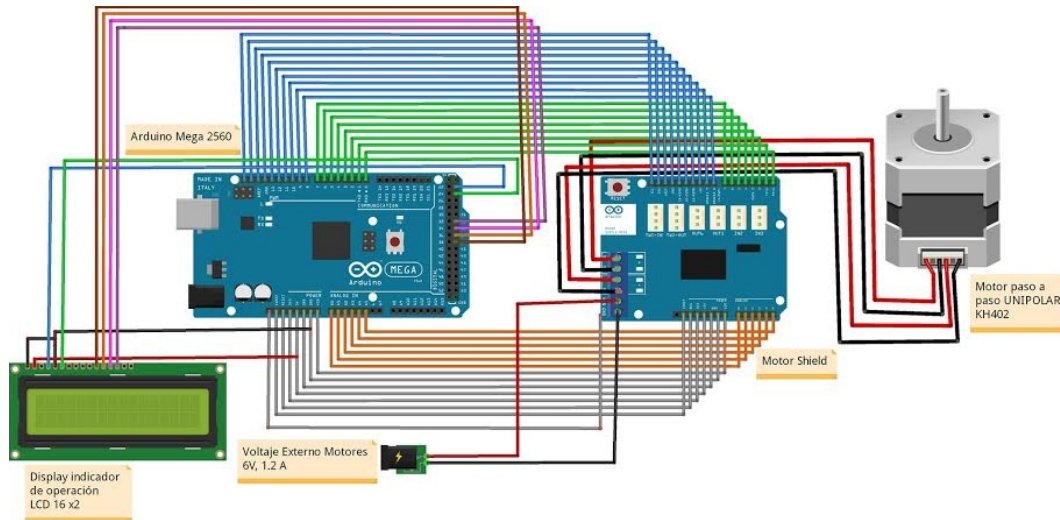


Figura 5: Diagrama de flujo, funcionamiento del sistema. Fuente: Elaboración propia.

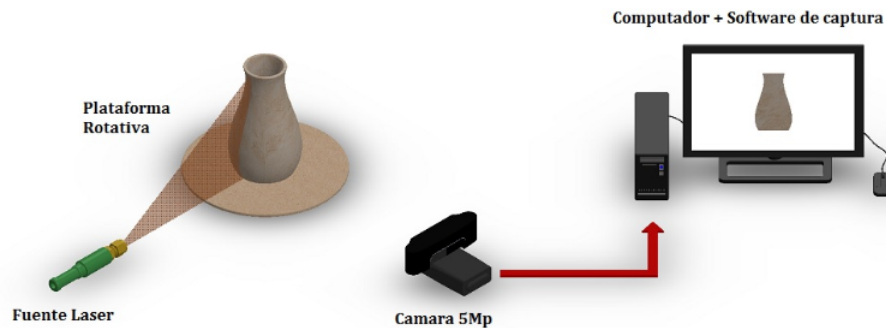


Figura 6: Escáner 3D con haz de luz láser. Fuente: Elaboración propia.

El algoritmo de reconstrucción usa un proceso de cambio de coordenadas, en el que las coordenadas cartesianas obtenidas de la imagen son convertidas a coordenadas cilíndricas (Sevilla, 2007), lo que permite visualizar la figura en un espacio euclidiano (Marrero, 2009).

En coordenadas cilíndricas, un punto P es representado por (r, θ, z) , como se muestra en la figura 7.

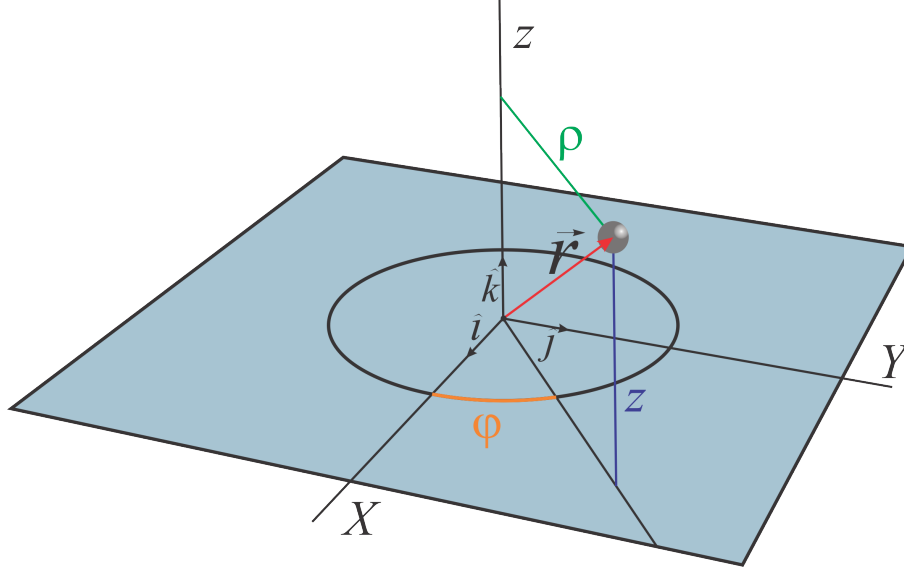


Figura 7: Coordenadas cilíndricas. Fuente: Elaboración propia.

La equivalencia entre las coordenadas cartesianas y las coordenadas cilíndricas está dada por las ecuaciones 1, 2 y 3:

$$x = \gamma \cos \theta \quad (1)$$

$$y = \gamma \sin \theta \quad (2)$$

$$z = z \quad (3)$$

donde:

- γ : Coordenada radial, es la distancia del punto P al eje z .
- θ : Coordenada azimutal, es el ángulo que se forma con el eje x .
- z : Coordenada vertical, es la distancia desde el punto P al plano XY .

En este trabajo, se toma como valor de r , la distancia con el eje y , para cada uno de los puntos (x, y) , como se muestra en la figura 8.

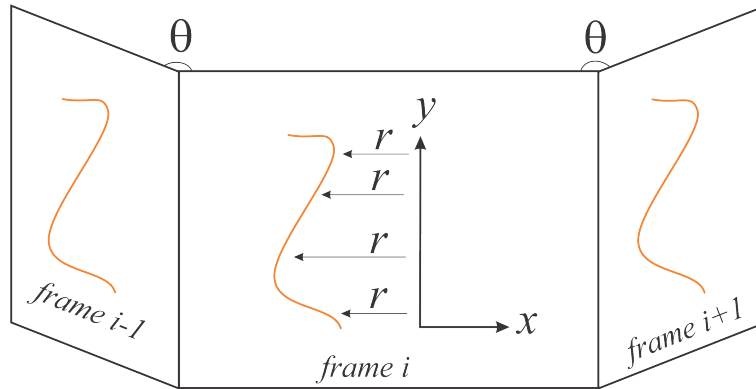


Figura 8: Reconstrucción tridimensional con escáner láser (Marreros, 2009).

El valor de θ , que indica el ángulo en el cual se dibujará la traza, es determinado dependiendo del número de cuadros (*frames*) que se presenten, como se muestra en la ecuación 4:

$$\theta = \frac{360}{\text{número de cuadros}} \quad (4)$$

5. RESULTADOS Y ANÁLISIS

Durante el desarrollo del proyecto, se evidenció que los modelos tridimensionales generados, presentaban un porcentaje de error, alrededor de 400 % como se muestra en la Tabla 1

Tabla 1: Medidas iniciales.

Ejes	Medidas reales (cm)	Medidas teóricas (cm)	Error(%)
x,z	3.6	17.80	394
y	7.7	38.74	403

La Figura 9 muestra los ejes que se tomaron para el ajuste del objeto virtual con el objeto real.

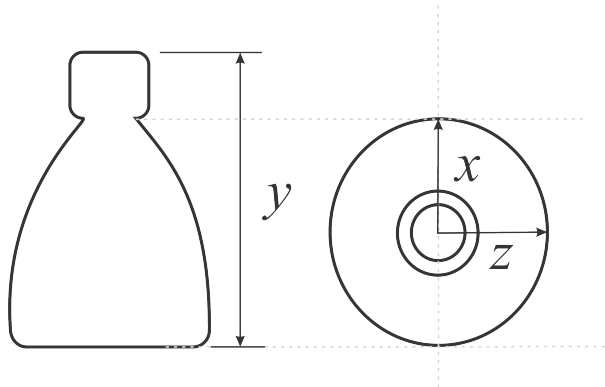


Figura 9: Toma de ejes para calibración del objeto. Fuente: Elaboración propia.

Las ecuaciones 5 y 6, fueron utilizadas para la alteración de los modelos futuros.

$$C_i = \frac{M_r}{M_t} \quad (5)$$

$$E_i = E_i \times C_i \quad (6)$$

donde:

- M_r : Medida real
- M_t : Medida teórica
- C_i : Constante de calibración
- E_i Ejes (z,y o z)
- i índice de cada eje

Los factores de escala obtenidos para cada uno de los ejes son los siguientes:

$$fx = 0.20224 \quad fz = 0.20224 \quad fy=0.19876$$

Una vez aplicados estos valores, los modelos presentaban los errores mostrados en la Tabla 2.

Tabla 2: Medidas finales.

Ejes	Medidas reales (cm)	Medidas teóricas (cm)	Error(%)
x,z	3.6	3.4	6
y	7.7	7.8	1

En la Figura 10, se puede observar algunos de los modelos obtenidos mediante el uso del escáner desarrollado. Los objetos reconstruidos reflejan la forma original de los objetos reales ofreciendo un acabado superficial rugoso a la vista debido a la construcción generada por el algoritmo de formación del objeto. Es evidente que se requiere un proceso de simplificación de nodos que permitan generar superficies lisas adaptadas a las necesidades de los usuarios. Las imágenes generadas presentan solapamiento de nodos que deben ser tratados con algoritmos para su corrección de manera que las imágenes puedan ser modificadas desde la malla de datos generada.

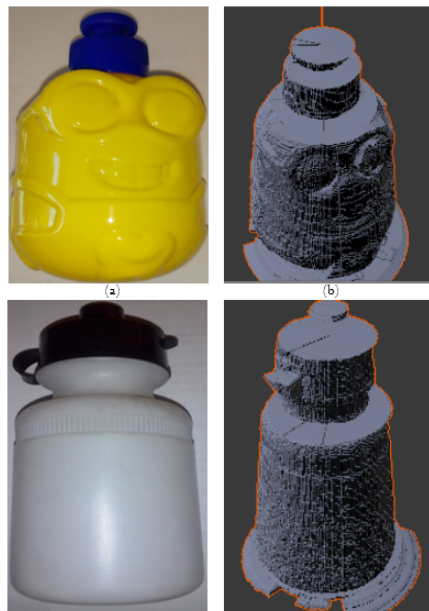


Figura 10: Objetos escaneados. (a) y (c) objetos originales, (b) y (d) imagen de objetos modelados correspondientes en entorno tridimensional. Fuente: Elaboración propia.

6. CONCLUSIONES

Durante el desarrollo de este proyecto se logró evidenciar que:

La reconstrucción de objetos reales en un entorno virtual puede emplearse para facilitar áreas de desarrollo como videojuegos o incluso en la preservación de piezas arqueológicas en delicado estado ya que se tiene una técnica de no contacto que facilita su digitalización.

Los conceptos de visión por computadora y procesamiento de imágenes, así como su aplicación en la vida real, pueden ser abordados con el uso de herramientas como las librerías OpenCV y OpenGL, las cuales permiten un ágil desarrollo de proyectos en los que se requiere el uso de estos tópicos.

El uso de hardware libre, como la plataforma de circuito programable Arduino, junto con sus diferentes shields, facilitan el desarrollo de proyectos, en los que se requiere la interacción entre hardware y software, además, de brindar flexibilidad a los mismos.

Los objetos generados pueden ser usados como elementos constituyentes de una escena tridimensional, pero en caso de que se requieran para un proceso de edición, se hace necesario el uso de un tratamiento de la malla de datos, esto con el fin de eliminar el solapamiento de nodos generado por el algoritmo de reconstrucción. Algoritmos como Ganapathy y Delauny, pueden ser implementados para este fin.

Referencias

- Arduino (2016). Obtenido de Arduino: <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>.
- Baldassari, S. (2013). Modelado geométrico. Obtenido de: <http://webdiis.unizar.es/sandra/masterig/modgeométrico13-14.pdf.4>
- Bradski, G. & Kaebler, A. (2008). Learning OpenCV: Computer Vision with the OpenCV Library. Beijing: O'Reilly Media; 1st edition.
- Carletti, E. J. (2007). Control de motores de cc puente h. *Robots Argentina*. Obtenido de: <http://robots-argentina.com.ar/MotorCCPuenteH.html>
- Conti, F. (2005). Motores paso a paso. Buenos Aires, Argentina: Librería y Editorial Alsina; 1 edition.
- Crespo, C. M & Fernández, E. (2014). Diseño e implementación de un escáner 3D para prototipado y modelado geométrico de objetos. Tesis Pregrado, Universidad Autónoma del Caribe.
- Galdames, O. (2011). Modelización con curvas y superficies de Bézier. *Modelling in Science Education and Learning*, 4(4), 181–193.
- Llamas, L. (2014). Tutoriales arduino, comunicación de arduino con puerto serie. Obtenido de: <http://www.luisllamas.es/2014/04/arduino-puerto-serie/>.
- Logitech, L. H. (2010). Webcam c525. Obtenido de HD Webcam C525: <https://neocenter.com/2015/img/logitech/176-85-LOCC525.pdf>.
- Logitech, L. (2016). HD WEBCAM C525. Obtenido de <http://www.logitech.com/es-roam/product/hd-webcam-c525>.
- Marrero, F. (2009). Escáner 3D mediante Triangulación y Luz Estructurada para Reconstrucción de Piezas Arqueológicas. Benemérita Universidad Autónoma de Puebla.
- Monk, S. (2012). Programming Arduino getting started with sketches. New York: McGraw-Hill.
- Palazzesi, L. (2008). Motores paso a paso. Obtenido de <http://www.neoteo.com/motores-paso-a-paso>.
- Salgado, D., Castro, L., & Castro, S. (2008). Subdivisión de superficies Beta-spline cúbicas. Proceedings XIV Congreso Argentino de Ciencias de la Computación, Oct. 2008.
- Scala Estalella, J. J. (2012). Análisis Vectorial, Volumen I: Vectores. España: Reverte ediciones.

Sevilla, L. D. (2007). Coordenadas cilíndricas. Definición. team, o. d. (2016). OpenCV. Obtenido de OpenCV: <http://docs.opencv.org/2.4/genindex.html>.