Reci
Revista
Iberoamericana de las
Ciencias Computacionales
e Informática
ISSN: 2007 - 9915

Prototipo para la automatización digital de objetos 3D apoyado en hardware y software libre

Prototype for the digital automation of 3d objects suported in hardware and free software

Protótipo para automação digital de objetos 3D suportados por hardware e software livre

Enrique Solis Cielo
Universidad Autónoma del Estado de México, México
e.solis92@hotmail.com

Oziel Lugo Espinosa
Universidad Autónoma del Estado de México, México
ozieluz@gmail.com

Joel Ayala de la Vega
Universidad Autónoma del Estado de México, México
joelayala2001@yahoo.com.mx

Alfonso Zarco Hidalgo
Universidad Autónoma del Estado de México, México
azarcox@hotmail.com

Vol. 6, Núm. 12 Julio – Diciembre 2017 DOI: 10.23913/reci.v6i12.72

Resumen

El prototipo presentado en este artículo es una combinación de hardware y software libre para la captura de imágenes en tres dimensiones (3D). Se da prioridad al uso de hardware libre y la reducción de costos, con el objetivo primordial de asegurar que el producto pueda ser desarrollado o adquirido por cualquier institución u organización, ya sea de investigación o desarrollo, de la forma más accesible.

Para lograr el prototipo, este fue subdividido en: la creación de un estudio fotográfico; la automatización del movimiento de la cámara utilizando dos motores a pasos y, como elemento de control, la placa Arduino Mega 2560; la selección del software que se encarga de generar el modelo 3D; y, por último, el desarrollo del software que permite la interacción de todos estos elementos.

Palabras clave: automatización, hardware libre, modelado 3D (3 dimensiones), sistemas embebidos, software.

Abstract

The prototype presented in this article is a combination of free hardware and software for the capture of images in three dimensions (3D). Priority is given to using free hardware and reducing costs, with the primary objective of ensuring that the product can be developed or acquired by any institution or organization, be it research or development in the most accessible way.

To achieve the prototype, it was subdivided into: the creation of a photographic studio; the automation of the movement of the camera using two stepper motors, and as control element the Arduino Mega 2560 board; the selection of software that is responsible for generating the 3D model; and, finally, the development of software that allows the interaction of all these elements.

Keywords: automation, free hardware, 3D modeling, embedded systems, software.

Resumo

O protótipo apresentado neste artigo é uma combinação de hardware e software livre para capturar imagens em três dimensões (3D). É dada prioridade ao uso de hardware e redução de custos gratuitos, com o objetivo principal de garantir que o produto possa ser desenvolvido ou adquirido por qualquer instituição ou organização, seja de pesquisa ou desenvolvimento, da maneira mais acessível.

Para alcançar o protótipo, foi subdividido em: a criação de um estúdio fotográfico; a automação do movimento da câmera usando dois motores passo a passo e, como elemento de controle, a placa Arduino Mega 2560; a seleção de software que é responsável por gerar o modelo 3D; e, finalmente, o desenvolvimento de software que permite a interação de todos esses elementos.

Palavras-chave: automação, hardware livre, modelagem 3D (3 dimensões), sistemas embarcados, software.

Fecha Recepción: Octubre 2016 **Fecha Aceptación:** Mayo 2017

Introduction

A 3D model is a digital representation of a real object that has the purpose of faithfully reproducing its physical characteristics, such as volume and form (Mesa Múnera, Ramírez Salazar and Branch Bedolla, 2010).

At present, the possibility of digitizing real-world objects in 3D allows the development of scientific, educational and entertainment applications that represent a powerful tool with a wide range of approaches for each of these. However, for many organizations, such as educational or scientific research, acquire a device that is able to recreate the structure of an object digitally represents a high cost investment that often can not afford.

There are different software alternatives on the market that allow carrying out 3D modeling tasks from different techniques. Some programs allow you to model from scratch and as a "drawing" objects that can be found in the real world and generate them digitally according to the creativity of the user. Others perform modeling from a physical object through techniques of obtaining and digitizing information for further processing.

To obtain a 3D modeling, we have the following techniques: those based on lighting use the projection of a beam of light or laser on the surface of the object to be digitized in order to highlight specific points of it (Mesa Múnera, Ramírez Salazar, & Branch Bedolla, 2010); other techniques are based on photogrammetry, which is the obtaining of real characteristics of an object from a set of photographic images of it from different perspectives (Rodriguez Jordana and Nuñez Andres, 2007); and, perhaps, one of the most complex techniques is the modeling by means of the use of ultrasonic sensors, which allow to determine the size and volume of an object (Orozco Quinceno, Romero Acero, Marín Cano, & Jímenez Builes, 2014).

When reviewing all the techniques, modeling was selected from photogrammetry, since the main objective of the project was the design and development of a low cost alternative that allows the modeling of an object in three dimensions and that is easy to use. any type of user. Since in case of failures in the system of movement and capture is much easier the acquisition and replacement of a webcam than an ultrasonic or infrared sensor, or even a laser. In addition, the processing is greatly simplified and avoids having to recalibrate the physical characteristics of each of these elements in each replacement.

Throughout this document we intend to address the process of design and development of this platform, starting with a brief introduction that tells us about the concept, as well as some

3D modeling techniques and applications. Subsequently, the base methodology that was used to streamline the design of this project and each of the phases of development of both modules individually is described. Finally, the full operation of the project is described once both parts of it are integrated, as well as some of the tests carried out on it are explained in order to verify its operation and performance in a real environment.

Methodology

For the development of this prototype, the embedded hardware and software development methodology proposed by Perez, Urkidi, Berreteaga, Ruiz de Olano and Pérez (2006) was used. This methodology was selected because it focuses on optimizing the design and development process of the project while allowing feedback and early evolution as deficiencies are identified in it.

The concept of embedded refers to hardware and software that work together as a whole and in an "invisible" way for the user. This methodology is known as Methodology V and consists of the following stages of development:

- Specification of HW / SW requirements. In this first stage of the methodology, it is necessary to define and document the different requirements that the system to be developed is intended to meet, identifying as much as possible the greatest number of specific numerical values.

- High level design. During this stage, we seek to obtain an overview of the functions of the system.

- Design in detail. It consists basically in detailing each of the functions defined in the previous stage.

- Implementation Here the design must be materialized in detail.

- Unit test. In this stage, each module that makes up the hardware and software part in a unified way is verified to measure its operation.

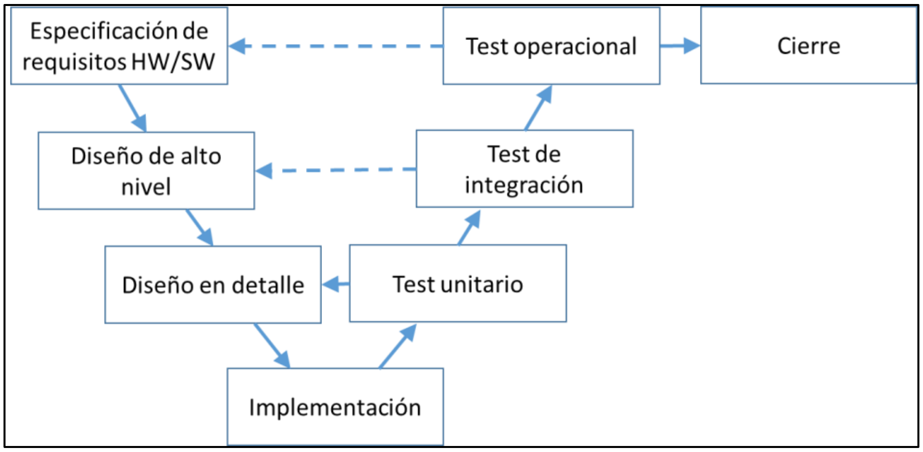
- Integration test. It consists, as its name indicates, in the total integration of the different modules that make up the system in order to check the correct functioning of the system and, in case of finding failures, make the appropriate corrections to correct the problem. In addition, compliance with the requirements established during the first and second stages must be verified.

- Operational test. The system is tested in a real environment and, if possible, in its final location.

- Closing. Delivery of the system.

Figure 1 shows the diagram that represents the life cycle of methodology V.

Figure 1. Life cycle of the methodology V.



Source: Perez *et al.* (2006).

1. The prototype

To develop the project, six elements were worked on separately:

- The creation of the photographic studio.
- The portability of the structure.
- The lighting system.
- The hardware that allows the movements and shots of the camera.
- The development of the software to be able to control the movements and photographs taken in the photo studio.
- The integration of software to model in 3D.

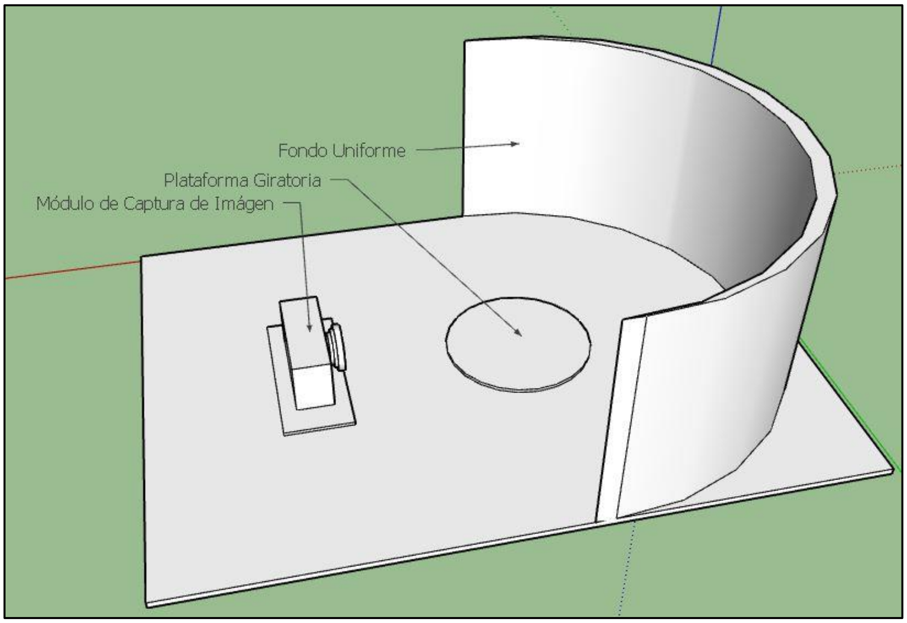
1.1 Creation of the photo studio

For the creation of the prototype of the photographic studio, where the digital images of the object to be modeled in 3D are taken, plywood was used and painted white in a matt tone in order to avoid reflecting light. On the lower face of the hub, a small groove was made through which the motor shaft responsible for the rotation of the platform passes. Similarly, on the left side of the structure another slot was made so that, as with the lower face, the motor shaft responsible for raising the position of the camera passed. The curved bottom is fastened to the lower and back side of the cube by means of velcro, which allows it to be easily assembled and disassembled, and the lighting system remains mobile in case it needs to be placed at a specific point that facilitates the illumination of the faces of the object.

1.2 Portability

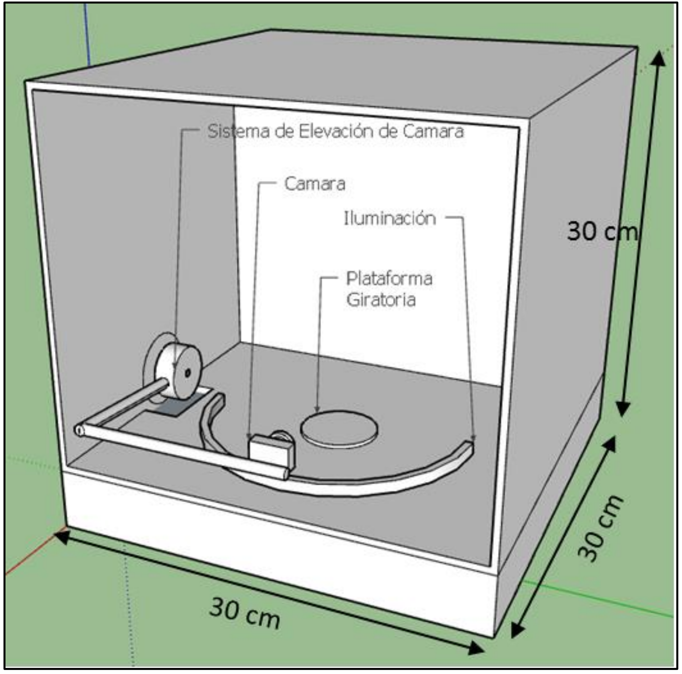
The first design of the structure was thought of as a semicontrolled environment, consisting of a half-cylinder design with a fixed support for capturing the photographs (see Figure 2). To facilitate the transportation of the prototype, it was changed to a cubical structure with a lower compartment to cover all the circuitry and allow the prototype to be assembled and disassembled in a short time without damaging any of the components. Figures 3 and 4 show the final structure of the prototype.

Figure 2. Estructura inicial del prototipo.



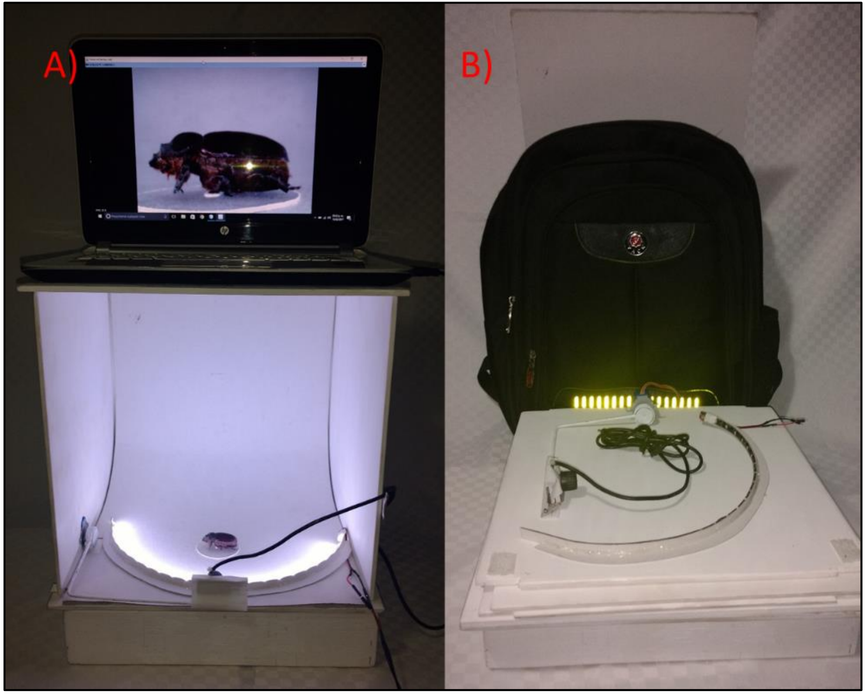
Fuente: elaboración propia.

Figure 3. Estructura final del prototipo.



Fuente: elaboración propia.

Figure 4. A) Estructura Final en funcionamiento B) Estructura final completamente desmantelada y lista para transportar en una maleta.



Fuente: elaboración propia.

1.3 Lighting frequency and environment

During the design of the prototype, two points were subjected to different tests in order to increase the performance of the same. The first point was the lighting system; Initially the tests were carried out by using an ordinary flash. However, it was detected that, when taking the photographs, this generated a shadow of the object at the bottom of the shot, which was more difficult for the modeling software to identify the volume and shape of the object. Subsequently, the flash was replaced by a constant lighting system based on LED strips of different types, some of the most powerful generated a black line effect on the shot, due to the low refresh rate of the camera frames. The frequency (or rate) of refreshment refers to how many images the camera can capture per second, and its unit of measurement are the Frames Per Second (FPS) that, related to the proposed problem, we could translate to Hertz (Hz). Therefore, if the LED strip has a frequency different from that of the camera, it will generate a black line effect due to the null synchronization between the light flashes and the camera's image capture (Vazquez, 2016).

To solve this problem, it was tested with low frequency LED strips until a similar one could be selected, which is why the 3M led strip at 20 Hz was chosen, thus avoiding this problem.

A second point that was subjected to tests is the environment of the shots, since it is intended to eliminate as much noise as possible in the images. It was decided to implement a controlled environment with a semi-circular background that allows incorporating an optical effect known as "infinite background", a technique that is commonly used in marketing for product photography and allows to simulate an endless background behind the object in each shot (Guasco, 2012).

1.4 The hardware for the photo studio

As for the movement system, two motors were used in steps 28BYJ-48, each one with its corresponding driver of type ULN2003A. These motors are common as key pieces in teaching small robots and home positioners programmed with Arduino. The technical characteristics with which said engine counts are (Prometec, 2017):

- Rated voltage between 5V and 12V.
- 4 phases of excitation..
- 50Ω resistance.
- Torque of 34 N / m.
- Approximate consumption of 55 mA.
- 8 steps per turn.
- 1/64 reducer.

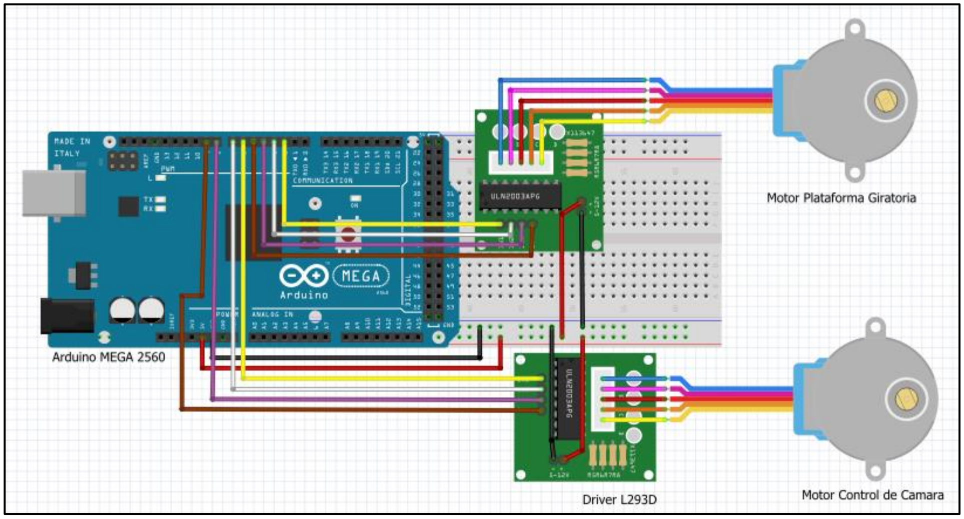
Translating the last two points in a more understandable way, we can deduce that multiplying the number of steps per turn (8) by the reduction index (64) results in a total of 512 electrical impulses to complete a full revolution of the motor.

The first motor was placed on the left side of the prototype and is responsible for calibrating the elevation of the camera to allow the capture of images at different angles of the object without losing stability. The second motor is placed in the lower central part of the prototype by means of a platform system where it allows the object to perform a 360 degree rotary movement on its vertical axis to capture the total contour of it. Both motors are controlled by means of the Arduino Mega 2560 board.

Arduino is a free hardware platform based on an input / output board (I / O), and a development environment that implements the Processing language (Fry and Reas, 2017). Arduino can be used to develop autonomous interactive objects or it can be connected via software with a computer. The platform can be assembled by hand or can be acquired already in use; The Integrated Development Environment (IDE) can be downloaded for free from the official Arduino website: www.arduino.cc (Banzi, 2011).

The Mega 2560 model is composed of a 16 MHz ATmega2560 microcontroller and a 256 KB flash memory. In addition, it provides a total of 54 digital input / output pins, and 16 analog inputs with their own development environment for programming the microcontroller (Perea, 2015). Your connection diagram is shown in the physical diagram (Figure 5).

Figure 5. Diagrama de conexiones físicas del circuito.



Source: elaboración propia.

Table 1 shows the unit cost of each element that makes up the proposed structure for this prototype.

Table 1. Costo unitario y total de producción del prototipo.

Elemento	**Costo Unitario**
Estructura de Triplay	$150.00
Placa Arduino Mega 2560	$740.00
Motor a pasos 28BYJ-48 (x2)	$150.00
Cámara web Ele-gate HD 18 Mpx	$180.00
PlastiAcero	$35.00
Pintura Acrilica Blanco Mate (x2)	$45.00
Driver Motor a Pasos (x2)	$124.00
Tira de Led	$30.00
TOTAL	$1,454.00 MN

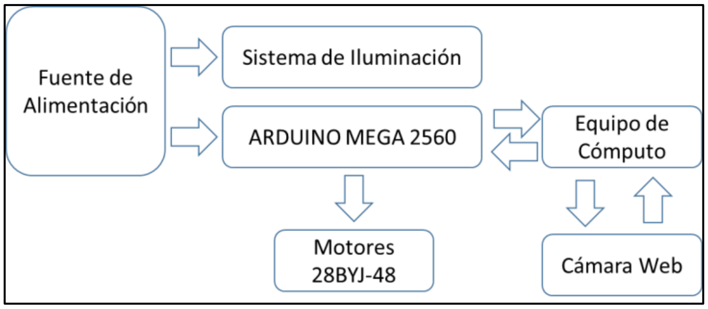
Source: elaboración propia.

There are different hardware alternatives designed to carry out the task of modeling. Some of the most economical examples that we can find for sale are the 3D "Matter and Form 3D" scanner of Matter and Form Inc. with an estimated price of $ 22,000 M.N. and the Sense handheld scanner developed by 3D Systems with a price of $ 15,000 M.N.

1.5 The software for photo studio control

The algorithm generated to carry out the control of the motors is based on simple instructions of repetitive blocks (loops) and in basic declarations of variables, in addition to the use of the analog inputs and outputs of the Arduino board (Torrente Artero, 2013). The operation of the electronic circuit is described in the block diagram of Figure 6.

Figure 6. Diagrama de bloques del circuito eléctrico.



Source: elaboración propia.

As for the software, a system mainly generated in Java was developed, which allows carrying out the calibration of the camera in real time and the capture of the final set of images automatically through synchronized control of both motors and direct communication with the camera.

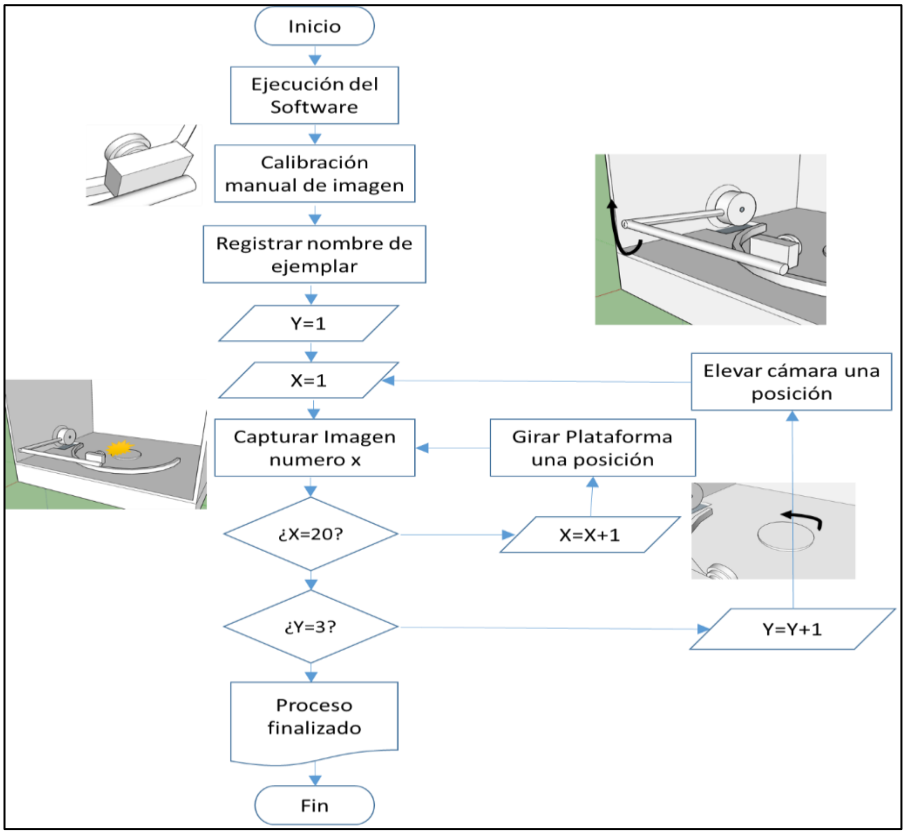
Finally, a modeling script known as MeshRecon is embedded within this software that allows, in a native way, to carry out the digitization task with the images obtained in the previous step.

The flow of processes carried out to capture all the images required for modeling is as follows:

1) Start the program execution.
2) Carry out manual calibration of the camera (focus and location).
3) Label example.
4) Set the start of the first session of three at a 0 ° angle.
5) Set the start of the first shot of 20 (1/20).
6) Capture the image.
7) If the image captured is the number 20/20, it advances to process 10, otherwise it advances to process eight.
8) Set the capture of the next image of the set of 20.
9) The object is rotated by an angle of 18 ° and process six is continued.
10) If the current session is 3/3, the capture process is finished, otherwise it goes to process 11.
11) Establish a new capture session.
12) Raise the camera at an angle of 20 ° with respect to its current position and continue in process number five.

Once the capture process is finished, the software continues with the processing of the obtained images so that from these it generates a 3D model of the object in question (see Figure 7).

Figure 7. Diagrama de flujo general del prototipo.



Source: elaboración propia.

1.6 The software for the treatment of images

There are commercial software to digitize objects in 3D, and ReMake® is one of them. He is the most recent successor of "123D Catch®". It was developed by the renowned company Autodesk. This software generates a 3D model of some real-world object by providing a set of between 50 and 120 photographs from different angles of the object in question and sending them through the Internet to the Autodesk servers through the same software application. ReMake® is a free license software, but it has paid extensions to increase its processing characteristics. ReMake® can be purchased from the Autodesk home page: https://remake.autodesk.com/about (Autodesk, 2015).

Another alternative as modeling software is PhotoScan®, product of the company Agisoft®, which is an advanced 3D modeling software, works locally on the computer and generates a digital representation of an object from photographs of it (Hagedorn- Saupe, Strawberry, Liestol, Rajcic and Grussenmeyer, 2016). Unlike ReMake®, PhotoScan® is a fully paid software with a 30-day trial period; After this period, the software license can be acquired from $ 3,205 M.N or a special license can be acquired, checking the educational purposes of the software from $ 1,056 M.N. PhotoScan® can be purchased from the Agisoft home page: http://www.agisoft.com/ (Agisoft, 2017)

These types of software are those that allow to use this prototype, since in a simple way it offers the ability to obtain a set of images with the appropriate characteristics to be processed later by some of them or by the modeling complement that contains in a native way this proposal.

Natively, the designed software works with a modeling script known as MeshRecon. This Script, proposed by Dr. Zhuoliang Kang (2015), is an adaptation of a software platform designed in 2012 for the modeling of 3D cities from aerial photographs, known as VisualSFM (Wu, 2014), which It is also a GUI (Graphic User Interface) with the ability to model structures from a trajectory of movement thanks to the implementation of multi-core parallelism technology incorporated by Nvidia graphics cards. The MeshRecon project can be consulted from the page: http://zhuoliang.me/meshrecon.html

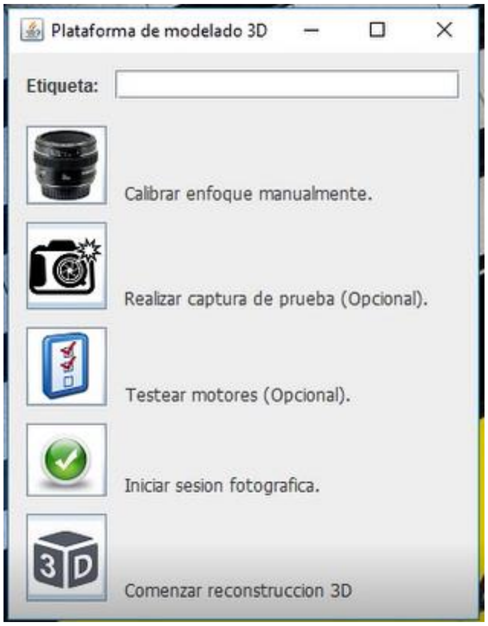
Results

Together, the platform works as an autonomous system that allows the capture of images for subsequent digital processing to a 3D model of it.

In the beginning, the software is automatically connected to the Arduino board, which in turn controls the motors. This system was developed in Java, a high-level programming language that has the characteristic of allowing the design and development of robust and portable software among various platforms of operating systems (Horstmann, 2010).

Initially, when executing the software, the main menu of the system will be displayed (see Figure 8); This menu shows the possible actions needed to successfully complete the modeling.

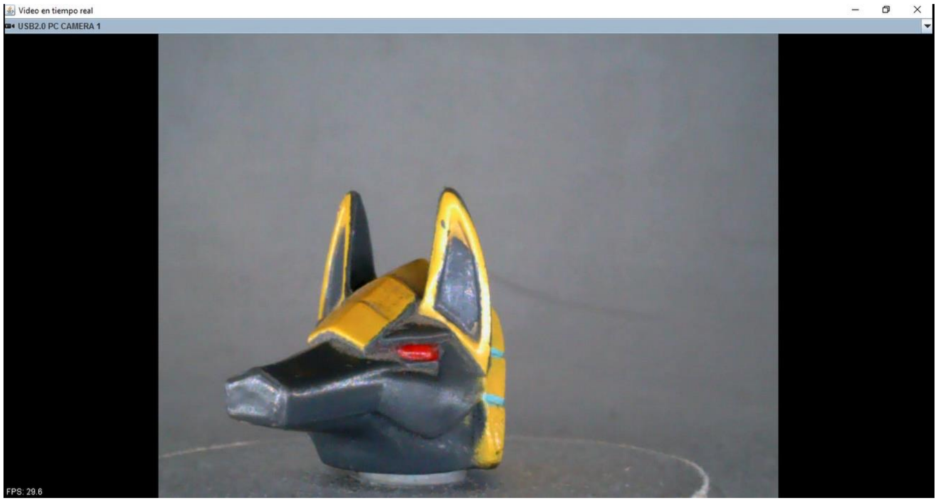
Figure 8. Menú principal del módulo de Software.



Source: elaboración propia.

The first button allows to open the calibration window for image focus (see Figure 9) by which the position and focus of the object is checked; the second button allows you to take a test photograph of the object; the third button begins the movement process of the motors simulating the capture of the photographs to verify that they work properly; a fourth button starts the process of capturing images for further processing and, finally, the fifth button allows you to begin the reconstruction of the object through the modeling script that is included natively.

Figure 9. Ventana de calibración de enfoque de la imagen.



Source: elaboración propia.

The platform captures a total of 60 photographs from different angles of the object, which are the ones recommended by the majority of 3D modeling alternatives from images.

Finally, the prototype was subjected to a series of tests in order to measure the time needed to capture images, and the processing time necessary to generate a 3D model. The results obtained indicate that the prototype takes approximately 2 minutes to take a total of 60 photographs. Each of these images has a storage weight of between 15 KB and 35 KB, giving an approximate average of 1.50 MB of storage for the final gallery that was later subjected to processing by means of different examples of modeling software.

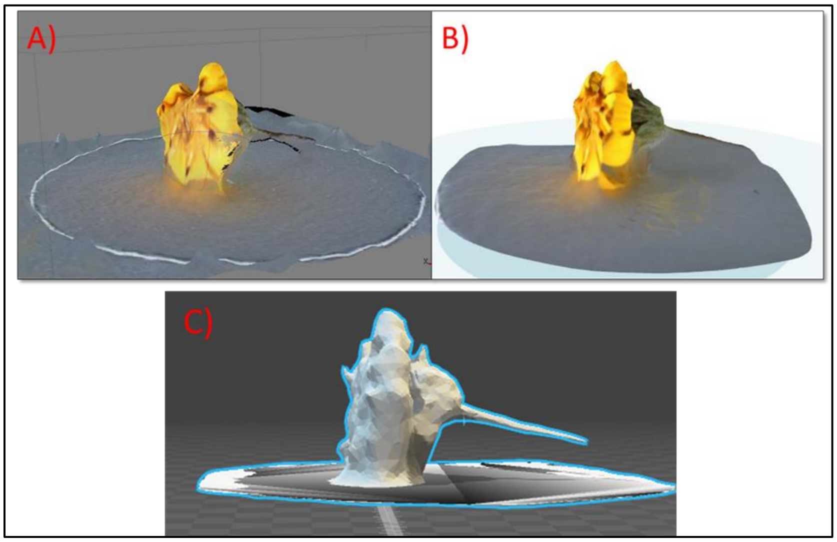
Below is a comparative table with the results obtained from two examples of plants and a plastic figurine that were subjected to the digitization process in which the differences in weight and processing times can be seen more clearly (Table 2) . In Figures 10, 11 and 12 an example obtained by each 3D modeling software is shown.

Table 2. Tabla comparativa de los resultados obtenidos a partir de pruebas con diferentes objetos.

Software	**Ejemplo**	**Tiempo de captura**	**Tamaño promedio de imagen**	**Tipo de procesamiento**	**Tiempo de procesamiento**	**Tipo de archivo final**	**Tamaño de archivo final**
PhotoScan	Flor Amarilla	2 Minutos	21.6 KB 640 x 480 px	Local	10 minutos	.psx	12.2 MB
	Flor Roja		25.8 KB 640 x 480 px		12 minutos		12.2 MB
	Figurilla de Plástico		22.1 KB 640 x 480 px		14 minutos		13.3 MB
ReMake	Flor Amarilla		21.6 KB 640 x 480 px	Remoto	15 minutos	.rcm	1.47 MB
	Flor Roja		25.8 KB 640 x 480 px		19 minutos		1.34 MB
	Figurilla de Plástico		22.1 KB 640 x 480 px		17 minutos		2.1 MB
MeshRecon (software nativo)	Flor Amarilla	2 Minutos	21.6 KB 640 x 480 px	Local	9 minutos	.ply	616 KB
	Flor Roja		25.8 KB 640 x 480 px		8 minutos		456 KB
	Figurilla de Plástico		22.1 KB 640 x 480 px		9 minutos		588 KB

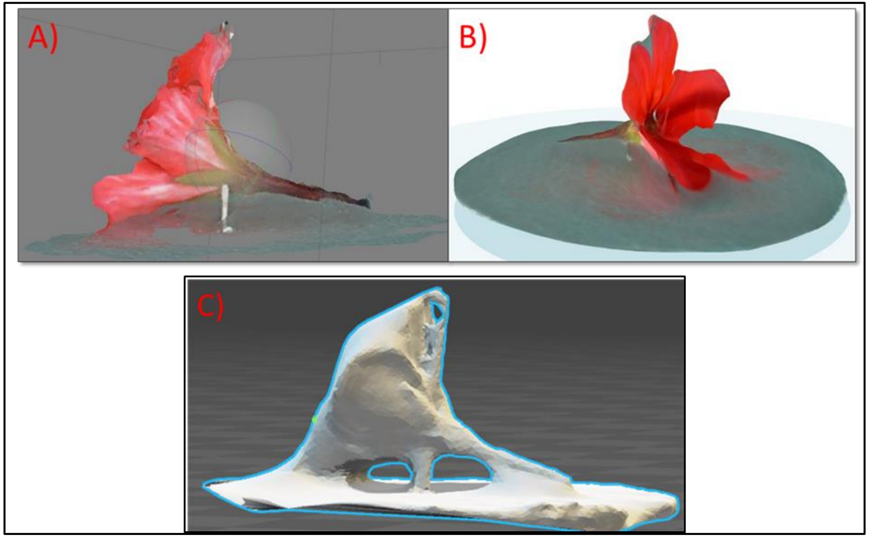
Source: elaboración propia.

Figure 10. Ejemplo 1 A) Resultado obtenido a partir de PhotoScan. B) Resultado obtenido a partir de ReMake C) Resultado obtenido a partir de MeshRecon.



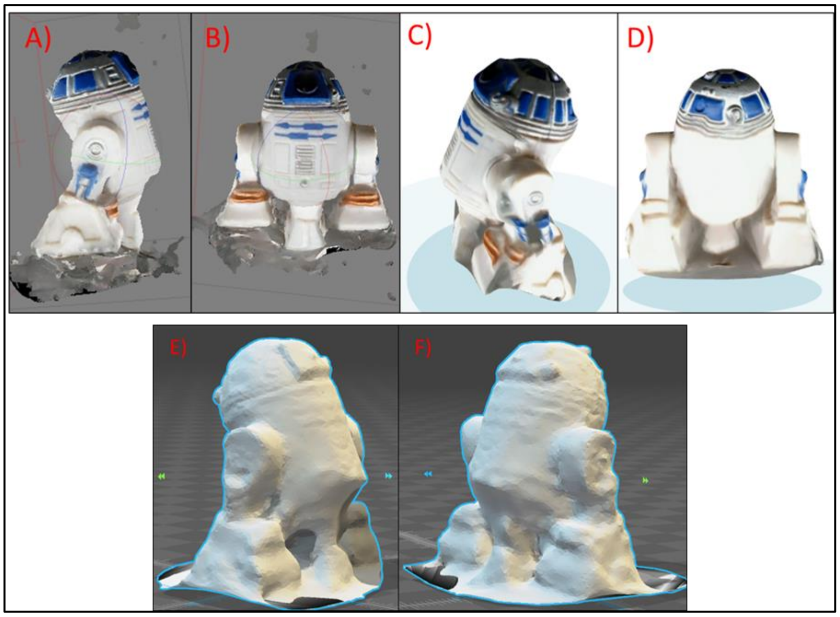
Source: elaboración propia.

Figure 11. Ejemplo 2 A) Resultado obtenido a partir de PhotoScan. B) Resultado obtenido a partir de ReMake C) Resultado obtenido a partir de MeshRecon.



Source: elaboración propia.

Figure 12. Ejemplo 3 A, B) Resultado obtenido a partir de PhotoScan (Vista lateral derecha y frontal). C, D) Resultado obtenido a partir de ReMake (Vista lateral izquierda y trasera) E, F) Resultado obtenido a partir de MeshRecon (Vista lateral derecha e izquierda).



Source: elaboración propia.

Taking into account that the upload speed at which the tests were performed at the time of uploading the images to the Autodesk server was 0.25 Mbps and, given that the total size of the gallery is 1.52 MB, the estimated upload time is of (1.52 / .25) 6.08 minutes. However, it is worth mentioning that the speed of the network may vary depending on the use of the network and the saturation of information of the same, so this time may be increased.

Discussion and Conclusions

With respect to the main objective of the project, this proposal proved to be a viable means to automate the process of digitizing 3D objects in a simple and intuitive way for any type of user and in a software environment that can be executed in different operating systems.

The final prototype has the facility to allow the structure to be assembled and untied with simplicity and speed. The transport of the structure is facilitated to avoid the acquisition of a copy of the prototype for each work space where it is required to carry out the digitalization, fulfilling the objective of portability.

The average response times to generate a complete digitalization are 15 minutes with 60 images and a final file weight between 1 MB and 12 MB depending on the software. This allows the user to obtain several models from the same set of photographs and select the one that best meets their expectations.

The current cost reduction compared to a commercial 3D scanner is approximately $ 13,500 MN compared to the Sense hand scanner from 3D Systems.

It should be mentioned that, for the current model of the prototype, the objects to be modeled can have a maximum size of 5 cm high and 5 cm wide. However, this size can be enlarged if the camera support moves further away from the center of rotation to expand the capture area.

Since Java allows the development and distribution of applications in the fields of mobile and embedded programming, video games, web content and business software and the execution of applications in various environments and operating systems, this contributes to the software created for this project also has the advantage of implementing portability between computing equipment with different technical characteristics.

Acknowledgment

We thank the National Council of Science and Technology (CONACYT) for the assistance provided, as well as the Autonomous University of the State of Mexico (UAEM), Texcoco University Center, for the support provided during the development of this research.

Bibliography

Agisoft. (2017). Agisoft. Obtenido de Agisoft: http://www.agisoft.com/

Autodesk. (2015). Autodesk ReMake. Obtenido de http://remake.autodesk.com/try-remake#system-requirements

Banzi, M. (2011). Getting Started with Arduino. O'Reilly Media, Inc., p. 2-6.

Fry, B., & Reas, C. (08 de 2017). Processing. Obtenido de https://processing.org/

Guasco, I. (2012). Tecnicas de Fotografia Profesional. Buenos Aires, Argentina: RedUSERS. pp. 32-38,112-116.

Hagedorn-Saupe, M., Fresa, A., Liestol, G., Rajcic, V., & Grussenmeyer, P. (2016). Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection. Nicosia, Chipre: Springer International Publishing. pp. 401-408.

Horstmann, C. (2010). JAVA For Everyone. Estados Unidos: John Wiley & Sons, Inc. pp. 6-11.

Kang, Z. (29 de 04 de 2015). Mesh Reconstruction from Imagery. Obtenido de http://zhuoliang.me/meshrecon.html

Mesa Múnera, E., Ramírez Salazar, J. F., & Branch Bedolla, J. W. (2010). Construcción de un modelo digital 3D de piezas precolombinas utilizando escaneo laser. Revista Avances en Sistemas e Informática 2010, 7(1), pp. 198-206.

Perea, F. (2015). Arduino Essentials. Birmingham, UK: Packt Publishing. pp. 11-12

Perez, A., Urkidi, A., Berreteaga, O., Ruiz de Olano, A., & Pérez, J. (2006). Una metodologia para el desarrollo de Hardware y Software embebidos en sistemas criticos de seguridad. Sistemas, cibernetica e informática., 3(2), pp. 70-75.

Prometec. (15 de 08 de 2017). Obtenido de Prometec: http://www.prometec.net/motor-28byj-48/

Rodriguez Jordana, J. J., & Nuñez Andres, M. A. (2007). Fotogametría arquitectónica. Barcelona, España: Ediciones UPC. pp. 23-30.

Torrente Artero, O. (2013). ARDUINO Curso Practico de Formacion. Madrid, España: RC Libros. pp. 60-71, 129-130.

Vazquez, A. (2016). La Frecuencia de refresco en las pantallas. Obtenido de http://ledandgo.com/la-frecuencia-de-refresco-en-las-pantallas.html

Wu, C. (2014). VisualSFM : A Visual Structure from Motion System. Obtenido de http://ccwu.me/vsfm/

Rol de Contribución	Autor(es)
Conceptualización	Enrique Solis <Principal>;Oziel Lugo <Apoya>
Metodología	Enrique Solis <Principal>
Software	Enrique Solis <Principal>; Oziel Lugo <Apoya>; Joel Ayala <Apoya>
Validación	Oziel Lugo <Principal>; Joel Ayala <Apoya>; Alfonso Zarco <Apoya>
Análisis Formal	Joel Ayala <Principal>; Alfonso Zarco <Apoya>
Investigación	Enrique Solis <Principal>
Recursos	Enrique Solis <Principal>; Oziel Lugo <Apoya>
Curación de datos	Enrique Solis <Principal>; Oziel Lugo <Apoya>
Escritura - Preparación del borrador original	Enrique Solis<Principal>
Escritura - Revisión y edición	Joel Ayala <Principal>; Oziel Lugo <Apoya>; Alfonso Zarco<Apoya>
Visualización	Enrique Solis<Principal>
Supervisión	Oziel Lugo <Principal>; Joel Ayala<Apoya>
Administración de Proyectos	Oziel Lugo <Principal>; Joel Ayala <Apoya>
Adquisición de fondos	CONACYT<Principal>; Enrique Solis<Apoya>