






Construcción e implementación de una aspiradora autónoma

Constructing and implementing of an autonomous vacuum cleaner

Cesar Marquínez¹, Gabriel Quintana², Kendrick
Sánchez³, Lilia Muñoz⁴, Vladimir Villarreal⁵

Marquínez, C; Quintana, G; Sánchez, K; Muñoz, L; Villarreal, V. Construcción e implementación de una aspiradora autónoma. *Tecnología en Marcha*. Vol. 36, número especial. Octubre, 2023. V Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software, Salud Electrónica y Móvil. Pág. 23-33.

 <https://doi.org/10.18845/tm.v36i8.6924>

- 1 Universidad Tecnológica de Panamá. Panamá.
Correo electrónico: cesar.marquinez@utp.ac.pa
 <https://orcid.org/0000-0003-2577-3221>
- 2 Universidad Tecnológica de Panamá. Panamá.
Correo electrónico: gabriel.quintana@utp.ac.pa
 <https://orcid.org/0000-0003-0189-0810>
- 3 Universidad Tecnológica de Panamá. Panamá.
Correo electrónico: kendrick.sanchez1@utp.ac.pa
 <https://orcid.org/0000-0003-1706-5460>
- 4 Universidad Tecnológica de Panamá. Panamá.
Correo electrónico: lilia.munoz@utp.ac.pa
 <https://orcid.org/0000-0002-4011-2715>
- 5 Universidad Tecnológica de Panamá.
Correo electrónico: vladimir.villarreal@utp.ac.pa
 <https://orcid.org/0000-0003-4678-5977>

Palabras clave

Microcontroladores; sensores; aspiradoras autónomas; electrónica; IoT.

Resumen

El objetivo de este proyecto es una implementación que conlleva diseño y desarrollo de una aspiradora de limpieza de bajo costos que realice su operación autónoma. La detección de obstáculos con la utilización de un sensor es la plataforma que permite la autonomía de la aspiradora de limpieza ya que usualmente este tipo de dispositivos necesitan del control y movimiento de un humano para su oficio. La noción de bajo costo es dada, ya que sus componentes son de precio accesible. La aspiradora autónoma de limpieza emplea sensor ultrasónico HC-SR04 que emite un ultrasonido que viaja por el aire y si hay un obstáculo en su camino podrá esquivar mediante la gestión de su comportamiento con Arduino UNO. Una aspiradora autónoma de limpieza nos da una visión general y de escalabilidad de cómo las herramientas de código abierto son aprovechadas para cimentaciones de proyectos electrónicos con miras a usos de estrategias de Internet de las Cosas (IoT) e Inteligencia Artificial.

Keywords

Microcontrollers; sensors; autonomous vacuum cleaners; electronics; IoT.

Abstract

The objective of this project is an implementation that entails the design and development of a low-cost vacuum cleaner that performs its autonomous operation. The detection of obstacles with the use of a sensor is the platform that allows the autonomy of the vacuum cleaner since these types of devices usually need the control and movement of a human for their job. The notion of low cost is given since its components are affordable. The autonomous vacuum cleaner uses an HC-SR04 ultrasonic sensor that emits an ultrasound that travels through air and if there is an obstacle in the way, it can avoid it by managing its behaviour through Arduino UNO. An autonomous vacuum cleaner gives us an overview and scalability of how open-source tools are leveraged for foundations of electronic projects and strategies' use of Internet of Things (IoT) and Artificial Intelligence.

Introducción

A finales de la década de 1990, surgió en California una nueva forma de entender el acceso a la tecnología. La idea principal era que aprendiésemos unos de otros en un entorno donde la información se comparte y está disponible en todo momento. Este ambiente inmersivo de tener cosas que puedan utilizarse y combinarse con tecnología ha permitido que las personas traten de facilitar sus vidas y quizás aun ciertas formas de trabajo las cuales necesiten mecanismos que puede ser de su vida cotidiana y que se operen por un humano. He ahí cuando se considera esa facilitación al tener artículos tradicionales que poseen cierto grado de autonomía con el manejo humano. Estos pensamientos de tener artículos del día a día con uso de la tecnología ha hecho posible la innovación y desarrollo de nuevos proyectos. Luego serán los usuarios de este movimiento quienes decidan cuales ideas son las mejores y también podrán realizar cambios para optimizarlas [1].

De esta forma, el intento de implementar sistemas autónomos requiere conocimientos interdisciplinarios, ya que comprenden algunos campos importantes como la mecánica, sensores, motores, comunicaciones, electrónica e inteligencia artificial. La base de esta investigación es hacer posible una fracción de la visión de un mundo en el que las máquinas nos asistan en la mayoría de los trabajos realizados por el hombre.

Es un hecho que las aspiradoras autónomas son un tema conocido, pues desde 2001 ya han aparecido varios modelos que se han desarrollado a la sombra del mercado [2]; y existen empresas que comercializan dichos ejemplares, como Roomba [3], robot aspirador Ubot [4], entre otras.

Dichos dispositivos comercializados mencionados anteriormente utilizan componentes de hardware y software, que sus costos son elevados, por ende, su adquisición es costosa. Por lo tanto, IRobots Roomba es una inspiración para diseñar y construir una aspiradora que esté disponible y que realice las funciones principales que ya existen en el mercado. Así nació el proyecto Hybrid-AV, que tendrá funciones de detección en un entorno limitado similares a Roomba.

En la figura 1, se muestra de manera generalizada conceptualizaciones de la aspiradora autónoma (Hybrid-AV) tales como: encendido y comienzo de recorrido, interceptación de obstáculos y detección de estos por medio de sensores, cambios de dirección, limpieza.

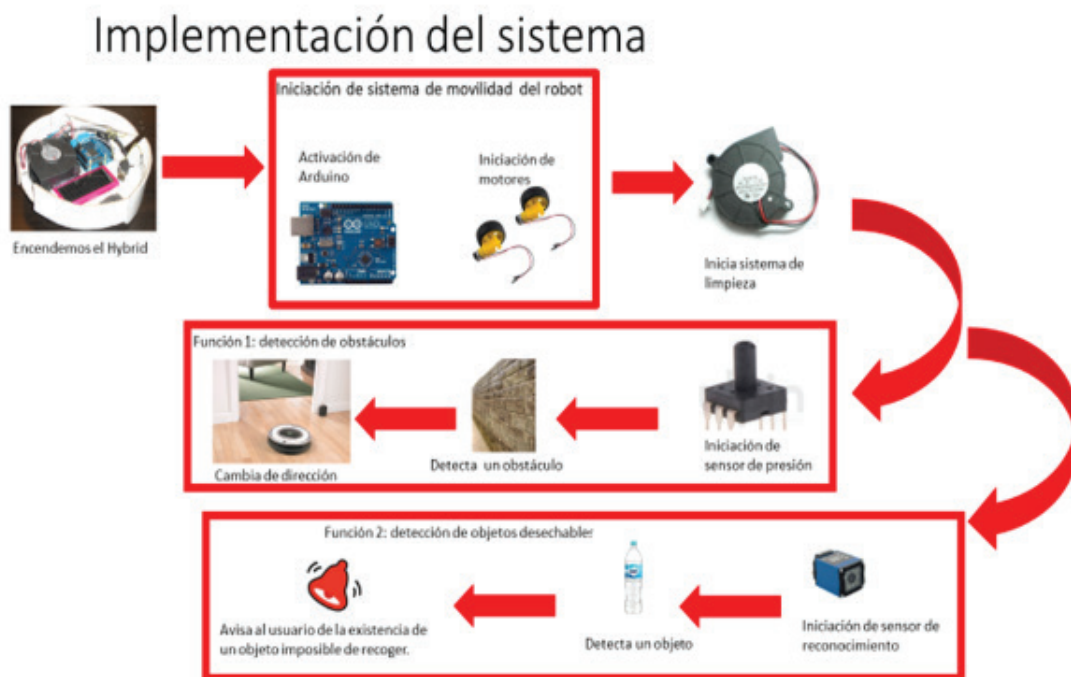


Figura 1. Conceptualización de la aspiradora.

Materiales y métodos

Para la descripción del diseño y metodología del desarrollo de este proyecto se requieren los siguientes materiales: Arduino UNO, Motor h-bridge con ruedas(x4) y ttv motor(x4), Baterías (12 volts), Ventilador (12 volts), Cables de conexión a USB, Cables de conexión a Arduino,

Goma, Plastic box -for dust-bin, Set sensores ultrasonido, Código de movimiento & detección de objetos, Lenguaje de programación *Phyton*, Programa de comparación de imágenes *Cascade trayner*

Metodología de diseño

En el desarrollo inicial, se diseñó un circuito lógico de los componentes en un simulador llamado Fritzing. Primero se conectaron los cables del Arduino UNO al protoboard seguidamente los motores de las ruedas para así proXbar su movilidad, se conectan los cables rojos y negros de los mismos como se muestra en la figura 2. Se colocan en los puertos del protoboard los cables del sensor ultrasonido, en los puertos negativo y positivo correspondientes para así tener la alimentación de energía en la misma. Al conectar todo en serie y paralelo se procede a conectar la fuente de energía en este caso las baterías de 12V. Se utilizará hardware y software libre como lo es la tecnología ARDUINO, el cual es el microcontrolador de este diseño. La aplicación de este microcontrolador nos proporciona la asistencia para poder emplear estrategias de programación, para crear ese flujo y navegación autónoma gracias a la ejecución de un algoritmo que permita que la aspiradora autónoma (Hybrid-AV) se desplace en un entorno desconocido. Se muestra en la figura 2 un circuito lógico del proyecto, y en la figura 3 un esquema físico del mismo:

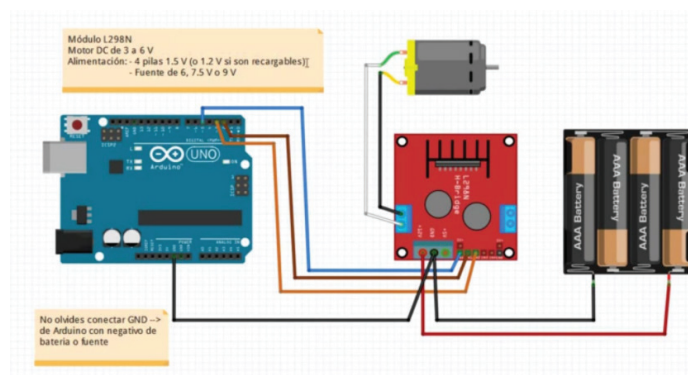


Figura 2. Circuito lógico de prueba de motores del Robot-aspiradora (Hybrid-AV).

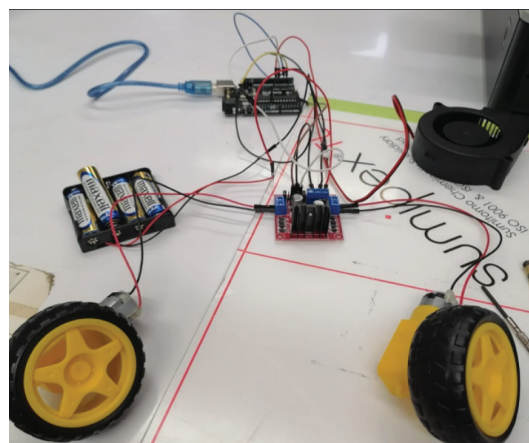


Figura 3. Conexión del circuito con los componentes.

Se realizan pruebas, de los códigos y el funcionamiento de los motores como se muestra en la figura 3. Se inicializa el armado en una base de cartón con un radio de 15cm, con 2 agujeros de forma centrada en la circunferencia de 9cm x 4cm para el espacio de las ruedas de los

motores, además 2 agujeros de un tamaño mínimo para pasar los cables del motor. Por otra parte, también se realiza un espacio para el ventilador, el cual se utiliza para realizar la limpieza. En la figura 4 se puede visualizar el ensamblado del proyecto.

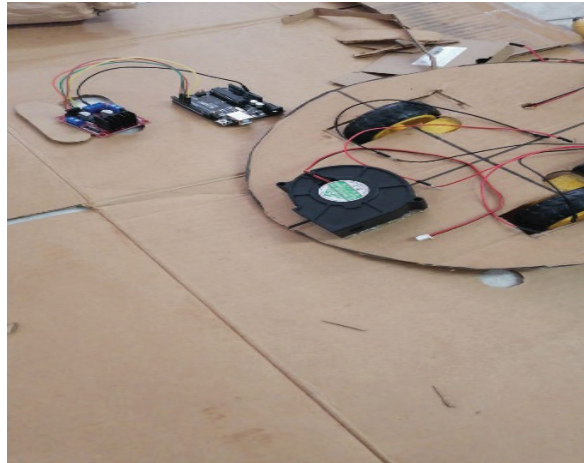


Figura 4. Estructura en proceso de armado.

Una vez fijados los componentes con una pistola de goma caliente, se procede a realizar las respectivas conexiones, en los puertos del controlador de motores (L298N) y el módulo de Arduino UNO.

Los motores se conectan a los puertos de disponibles en el controlador positivo y negativo (no importa el orden, nada más tenerlo en cuenta a la hora de realizar la programación en el código), una vez realizado esto conectamos nuestra fuente de voltaje de 5v (puede ser de 9v, o más, nada más tener en cuenta el funcionamiento y voltaje que permite el controlador de motores que se use).

La conexión se realiza de la siguiente forma: el positivo del cable de fuente (+) va con una unión en el primer puerto de entrada del controlador y el Vin (voltaje de entrada del Arduino) y cable negativo (-) iría al segundo puerto (o al puerto del medio) de la entrada del controlador y al GND (tierra) del Arduino.

Ahora los pines del controlador son 8 pero los de los extremos se usarán con puentes (o jumper) que trae el controlador (impórtate fijarse las placas de metal de los jumpers estén mirando hacia los pines), para el proyecto se usarán 4, el primer pin va a ir al ping 7 de Arduino UNO, el segundo ping va a ir al ping 6 de Arduino, el tercer ping va a ir al ping 5 de Arduino UNO, y por último el cuarto ping va a ir al ping 4 de Arduino UNO.

Por otra parte, se utiliza un sensor ultrasónico HC-SR04, el cual dispone de 4 salidas con sus respectivos colores, el cable rojo ira al tercer puerto de entrada del controlador de motores, y cable verde iría al ping A0 de la parte análoga del Arduino UNO, el cable azul iría al ping A1 de la parte análoga del Arduino UNO, y por último el cable negro iría al GND de POWER del Arduino UNO como se muestra en la Figura 5.

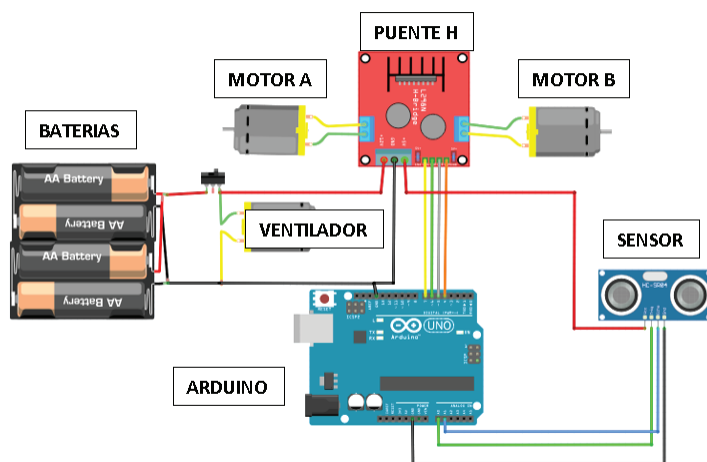


Figura 5. Circuito lógico del funcionamiento completo del Robot-aspiradora (Hybrid-AV).

Finalmente, una vez terminada esta parte de conexiones, se fijan los componentes y se prepara el código en Arduino UNO para cargarlo al módulo y realizar la prueba como se muestra en la figura 6.

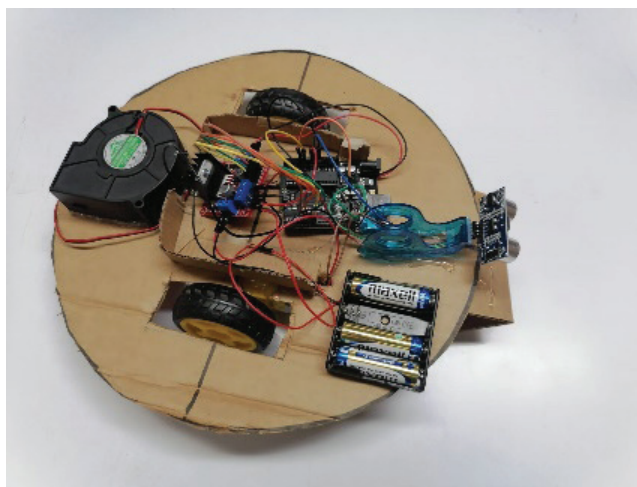


Figura 6. Prototipo armado con circuitos y componentes.

En la fase intermedia de prueba del prototipo, se realizaron mejoras o implementaciones como: el corte de la base de acrílico de radio de 15 cm para armar el equipo y organizarlo imitando la estructura de una aspiradora comercial. Como se muestra en la figura 7, se cambia la base del proyecto de cartón a plancha de acrílico:



Figura 7. Implementación de la base de acrílico.

Adicionalmente, se utiliza una estrategia de visión computacional utilizando reconocimiento de objetos. Esto se realiza para la detección de desechos que la aspiradora autónoma no pueda recolectar. Dicha estrategia utiliza código de lenguaje *Python* que junto a la librería *CV2* provee infraestructura para la percepción de imágenes, la cual puede hacer una comparación de los objetos detectados en el entorno sin mostrar falsos positivos, el mismo lo realiza utilizando un archivo generado por el programa *Cascade-Trayner*, llamado *cascade.xml*, se encarga de lo anteriormente mencionado.

Este entrenamiento se realizó tomando fotos del recuadro donde se localiza el objeto, en este caso unas fotos del *mouse* que se desea tomar como ejemplo para el reconocimiento. Lo mencionado anteriormente es una añadidura con usos de inteligencia artificial basadas fuera de la aspiradora autónoma.

Sin embargo, se encontraron limitantes que se tuvieron a la hora de implementar el diseño, las cuales se mencionan a continuación:

- Limitaciones en la integración de las cámaras para la detección de objetos.
- El procesamiento de imágenes no podrá ser utilizado con un Arduino UNO, ya que no lo soporta; no obstante, implementado un circuito aparte con una Raspberry pi se puede lograr.

Resultados

Se realizaron distintas pruebas con la implementación de los sensores, en el cuadro 1 se pueden apreciar los valores. Se demuestra que seis de diez pruebas realizadas el prototipo no se chocaba con los obstáculos que se le implementaron y el mismo cambiaba de rumbo.

Cuadro 1. Prueba del prototipo utilizando los sensores.

Pruebas de limpieza		
	Choque	No choque
Prueba 1		1
Prueba 2	1	
Prueba 3		1

Pruebas de limpieza		
	Choque	No choque
Prueba 4		1
Prueba 5		1
Prueba 6	1	
Prueba 7	1	
Prueba 8		1
Prueba 9		1
Prueba 10	1	

Como se muestra en la figura 8 se observa el prototipo en movimiento realizando la limpieza por medio de la aspiradora, el cual iba realizando el recorrido sin chocar con ningún obstáculo.

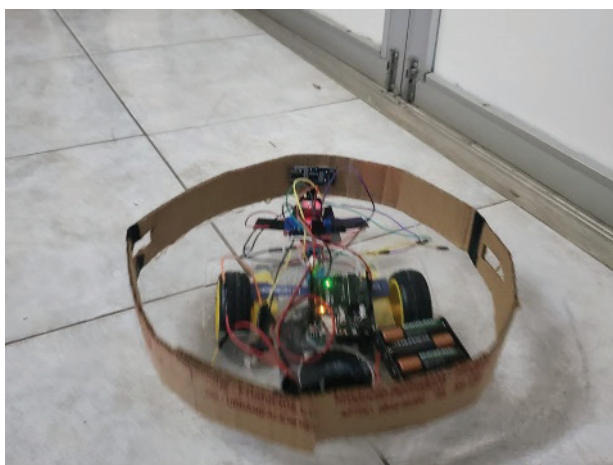


Figura 8. Prueba de movimiento del prototipo en área libre.

Una restricción que se presentó fue no poder montar la añadidura (uso de visión computacional para la detección de desechos que no podían ser recolectados) a la aspiradora autónoma por falta de componentes. No obstante, se realizaron pruebas de esta estrategia. En la figura 9 se observa uno de los objetos que se utilizó como referencia para la comprobación del código que se desarrolló.

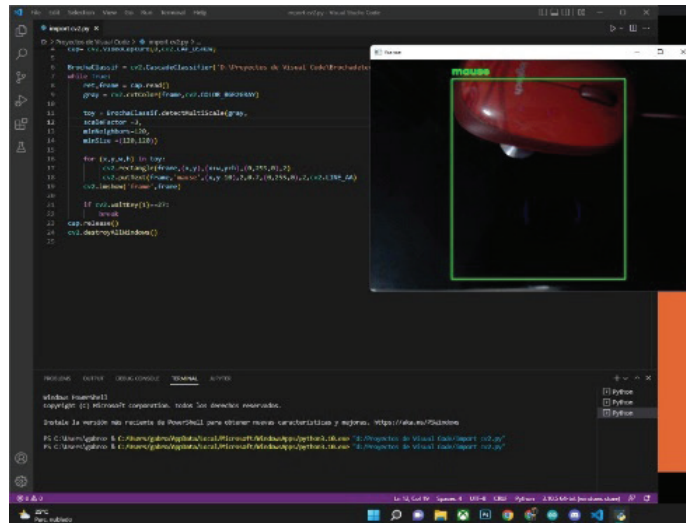


Figura 9. Implementación de la visión computacional.

Conclusiones

El objetivo principal del proyecto Hybrid-AV es obtener un prototipo de robot aspirador que se moviera de forma autónoma, adicionalmente se le implementaría una cámara para la detección de objetos que no pudiera limpiar y recolectar, elaborado con productos de código abierto y de la forma más económica posible. El resultado final ha sido satisfactorio, obteniendo las siguientes conclusiones:

- El prototipo se ha desarrollado con productos código abierto siendo así de fácil acceso para cualquiera.
- Se ha construido un prototipo funcional capaz de evitar obstáculos de forma autónoma e ir realizando el proceso de aspirar automáticamente.
- En base a que fuese lo más ecológico y sostenible posible, se ha usado baterías recargables y cartón para las fases de prueba.

En comparación a las aspiradoras que se encuentran en el mercado, que van desde los 150 dólares en adelante, con un sistema parecido, la idea de nuestro prototipo llegaría aproximadamente a los 100 dólares, cumpliendo con un auto recorrido completo, limpiando el entorno en donde se deje, con reconocimiento de objetos que no se pueden recoger, y sería una de las formas más económicas para realizar un proyecto de este tipo.

Agradecimientos

A la Secretaria Nacional de Ciencia, Tecnología e Innovación de Panamá, por el financiamiento parcial para el desarrollo del proyecto. La Dra. Lilia Muñoz y el Dr. Vladimir Villarreal son miembros del Sistema Nacional de Investigación de la SENACYT.

Referencias

- [1] OpenSource, «Open-Source Initiative, » [En línea]. Available: <https://opensource.org/history>.
- [2] Noticias De La Ciencia, «NACYT,» 08 octubre 2018. [En línea]. Available: <https://noticiasdelaciencia.com/art/30195/la-evolucion-de-los-robots-aspiradores#:~:text=Y%20todo%20comenz%C3%B3%20en%201996,idea%20del%20inventor%20James%20Dizon>.



- [3] IRobot, «Roomba,» [En línea]. Available: <https://www.irobot.es/roomba>.
- [4] Ubotcleaner, «Ubot,» [En línea]. Available: <http://www.ubotcleaner.com/>.
- [5] E. J. S. Paucar, «dspace,» 2009. [En línea]. Available: <http://dspace.esPOCH.edu.ec/bitstream/123456789/160/1/38T00152.pdf>.
- [6] J. M. A. Weber, «e-archivo,» 2018. [En línea]. Available: https://e-archivo.uc3m.es/bitstream/handle/10016/29335/TFG_Sandra_Ramos_Gutierrez.pdf;jsessionid=5F3817FC0750B31A8108289DA5DC3A7C?sequence=1.
- [7] M. A. S. Muñoz, «openaccess,» 21 Junio 2014. [En línea]. Available: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/35181/5/msanchezmunoz0TFC0714memoria.pdf>.
- [8] Revrekad, «instructables,» 29 ene 2015. [En línea]. Available: <https://www.instructables.com/Another-cardboard-robot-vacuum-cleaner-controlled-/>.
- [9] «amen-technologies,» 2022. [En línea]. Available: <https://es.amen-technologies.com/diy-smart-vacuum-cleaning-robot-using-arduino>.
- [10] J. Sanchez, «scribd,» 08 Apr 2021. [En línea]. Available: <https://es.scribd.com/document/502009361/Dialnet-PropuestaDePrototipoDeRobotAspiradoraDeBajoCostoYA-7563019>.
- [11] SoloElectronicos, «soloelectronicos,» 10 mayo 2016. [En línea]. Available: <https://soloelectronicos.com/2016/05/10/robot-aspirador-con-arduino/>.
- [12] ASSC, «assc.es,» [En línea]. Available: <https://assc.es/arduino-robot-aspirador-mejores/>.
- [13] T. X. Z. Avila, «tinkercad,» 23 julio 2021. [En línea]. Available: <https://www.tinkercad.com/things/iz5ekzbacza-aspiradora-1>.
- [14] «RN-XV WiFly modulo - wire antenna,» botnroll.com. [Online]. Available: <https://www.botnroll.com/en/ethernet-wi-fi/415-rn-xv-wifly-modulo-wire-antenna.html>. [Accessed: 09-Jun-2022].

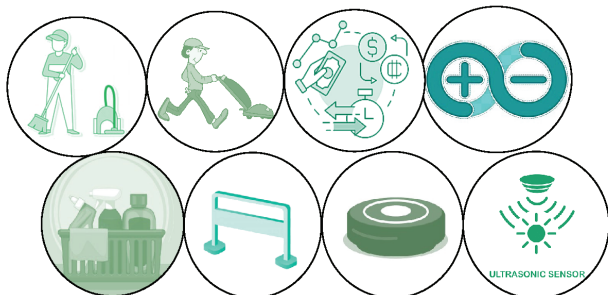


Construcción e Implementación de una Aspiradora Autónoma



Alexis Jiménez, Cesar Marquínez, Gabriel Quintana, Kendrick Sánchez

Licenciatura en Ingeniería de Sistemas y Computación, Facultad de Ingeniería de Sistemas Computacionales, Universidad Tecnológica de Panamá



RESUMEN

El objetivo de este proyecto es una implementación que conlleva diseño y desarrollo de una aspiradora de limpieza de bajo costos que realice su operación autónoma. La aspiradora emplea sensor ultrasónico HC-SR04, que emite un ultrasonido que viaja por el aire y en el momento que intercepta un obstáculo en su camino lo podrá esquivar mediante la gestión de su comportamiento con Arduino UNO.

INTRODUCCIÓN

Actualmente se cuentan con aspiradoras en el mercado pero las mismas son costosas y cumplen con algunas funciones limitadas por esta razón se realiza la propuesta de confeccionar una aspiradora robot de bajo costo con añadido de inteligencia artificial de detección de objetos. Esto es debido a que hay desechos que las aspiradoras no pueden recolectar y este añadido detectara e informará al usuario sobre los mismos.

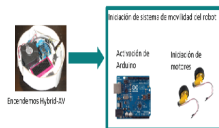
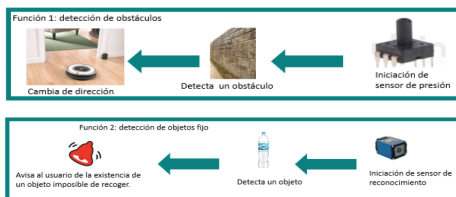


Diagrama de Conceptualización

PROPUESTA



DISEÑO Y METODOLOGÍA

Se conectan los cables del Arduino UNO al protoboard seguidamente los motores de las ruedas para así probar su movilidad. Se colocan en los puertos del protoboard los cables del sensor ultrasónico. Se conecta la fuente de energía en este caso las baterías de 12V. Se utilizará hardware y software de Arduino UNO. Se carga la programación al Arduino UNO para la navegación autónoma (Hybrid-AV). Para la creación del algoritmo de detección de objetos se utilizo la librería OpenCV y el programa Cascade Trainer para entrenar el mismo.

MATERIALES

Software

- Código de movimiento
- Código de detección de objetos
- Lenguaje de programación Phytton
- Programa de comparación de imágenes Cascade trayner
- Librería OpenCV
- Visual Studio Code (Editor de código)

Hardware

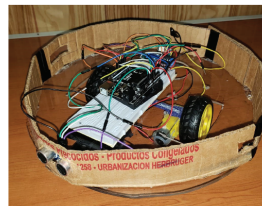
- Arduino UNO
- Motor h-bridge con ruedas(x4)
- Baterías (12 volts)
- Ventilador (12 volts)
- Cables de conexión a Arduino
- Goma
- Plastic box -for dust-bin
- Set sensores ultrasónico



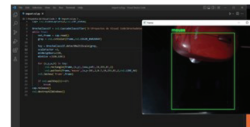
RESULTADOS

Se realizaron distintas pruebas con la implementación de los sensores en las cuales se recopiló la siguiente tabla. En ella se demuestra que seis de diez pruebas realizadas el prototipo no se chocaba con los obstáculos que se le implementaron y el mismo cambiaba de rumbo.

	PRUEBAS DE LIMPIEZA	
	CHOQUE	NO CHOQUE
PRUEBA 1		1
PRUEBA 2	1	
PRUEBA 3		1
PRUEBA 4		1
PRUEBA 5		1
PRUEBA 6	1	
PRUEBA 7	1	
PRUEBA 8		1
PRUEBA 9		1
PRUEBA 10	1	



Una restricción que se presentó fue no poder montar la añadidura a la aspiradora autónoma por falta de componentes. No obstante, se realizaron pruebas de esta estrategia.



CONCLUSIONES

El objetivo principal del proyecto (Hybrid-AV) era obtener un prototipo de robot aspirador que se moviera de forma autónoma, adicionalmente se le implementaría una cámara para la detección de objetos que no pudiera recolectar, utilizando código abierto. El resultado final cumplió en gran parte con lo propuesto, se ha construido un prototipo funcional capaz de evitar obstáculos de forma autónoma e ir realizando su proceso. No obstante, se han encontrado algunas restricciones. Debido a la falta de tiempo, no se logro finalizar un prototipo con la integración de las cámaras para la detección de objetos que el mismo no pudiera recolectar. Esta complicación se presente al montar el código utilizando Python debido a que Arduino UNO no soporta el procesamiento de imágenes sin embargo con la adaptación de una Raspberry pi sería posible.