



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO®

Instituto Tecnológico de Apizaco

---

**TECNOLÓGICO NACIONAL DE MÉXICO**  
INSTITUTO TECNOLÓGICO DE APIZACO

**PROGRAMA DE SEGUIMIENTO DE OBJETOS UTILIZANDO  
UNA CÁMARA PIXYCAM**

**TITULACIÓN INTEGRAL INFORME DE RESIDENCIA PROFESIONAL**

**PARA OBTENER EL TÍTULO PROFESIONAL DE:  
INGENIERA EN TECNOLOGÍAS DE LA INFORMACIÓN Y  
COMUNICACIONES**

**PRESENTA:  
ZAIRA HAZEL CRUZ BREMONT**

**NOMBRE DEL ASESOR:  
JOSÉ FEDERICO RAMÍREZ CRUZ**



**APIZACO, TLAX., OCTUBRE 2025**



Instituto Tecnológico de Apizaco



**EDUCACIÓN**



TECNOLÓGICO  
NACIONAL DE MÉXICO

Instituto Tecnológico de Apizaco  
Subdirección de Computación

Apizaco, Tlaxcala, 15 de noviembre de 2022.

OFICIO No: SC/OPV/189/22

**ASUNTO:** Liberación de proyecto  
para titulación integral

**ENRIQUE ACOLTZI BAUTISTA**  
**JEFE DEL DEPTO. DE DIVISIÓN DE ESTUDIOS PROFESIONALES**  
**P R E S E N T E.**

POR ESTE MEDIO LE INFORMO QUE HA SIDO LIBERADO EL SIGUIENTE PROYECTO PARA LA TITULACIÓN INTEGRAL

|                             |                                                                  |
|-----------------------------|------------------------------------------------------------------|
| <b>NOMBRE DEL EGRESADO:</b> | ZAIRA HAZEL CRUZ BREMONT                                         |
| <b>CARRERA:</b>             | INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES     |
| <b>NO. DE CONTROL:</b>      | 16371136                                                         |
| <b>NOMBRE DEL PROYECTO:</b> | PROGRAMA DE SEGUIMIENTO DE OBJETOS UTILIZANDO UNA CÁMARA PIXYCAM |
| <b>PRODUCTO U OPCIÓN:</b>   | TITULACIÓN INTEGRAL INFORME DE RESIDENCIA PROFESIONAL            |

AGRADEZCO DE ANTEMANO SU VALIOSO APOYO EN ESTA IMPORTANTE ACTIVIDAD PARA LA FORMACIÓN PROFESIONAL DE NUESTROS EGRESADOS.

**ATENTAMENTE**

Excelencia en Educación Tecnológica®  
Pensar para Servir, Servir para Triunfar®

**SAÚL OLAF LOATZA MELÉNDEZ**  
**JEFE DEL DEPTO. DE SISTEMAS Y COMPUTACIÓN**

c.c.p. División de Estudios Profesionales  
c.c.p. Depto de Servicios Escolares  
c.c.p. Archivo  
SOLM/mpm\*



**JOSÉ FEDERICO RAMÍREZ CRUZ**  
NOMBRE Y FIRMA DEL  
ASESOR INTERNO



Consejo Apizaco Tecnológico de Apizaco, Tlaxcala, 15 de noviembre de 2022.  
Consejo Apizaco Tecnológico de Apizaco, Tlaxcala, 15 de noviembre de 2022.  
c.c.p. multi (SOLM/mpm\*)



**2022**  
Ricardo Flores  
Magon



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

Instituto Tecnológico de Apizaco



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

Instituto Tecnológico de Apizaco  
División de Estudios Profesionales

Apizaco, Tlax., 22/febrero/2024  
**No. OFICIO: DEP-CT/208/2024**

**ASUNTO: AUTORIZACIÓN DE IMPRESIÓN**

**ZAIRA HAZEL CRUZ BREMONT**  
**EGRESADA DE LA CARRERA DE**  
**INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES**  
**CON NÚMERO DE CONTROL: 16371136**  
**P R E S E N T E**

Por este medio, me permito informar a usted que por aprobación de la comisión revisora asignada para valorar el trabajo que mediante la opción **TITULACIÓN INTEGRAL INFORME DE RESIDENCIA PROFESIONAL** cuyo tema es: **"PROGRAMA DE SEGUIMIENTO DE OBJETOS UTILIZANDO UNA CÁMARA PIXYCAM"**, conforme a lo establecido en el procedimiento para la obtención del Título Profesional en el Instituto Tecnológico, la División de Estudios Profesionales a mi cargo le otorga:

**AUTORIZACIÓN DE IMPRESION**

Debiendo entregar **3** empastados del mismo, apropiadamente encuadernados, a la brevedad, para presentar su acto de recepción profesional.

Sin otro particular por el momento, le envió un cordial saludo.

**ATENTAMENTE**

Excelencia en Educación Tecnológica.  
Pensar para Servir, Servir para Triunfar®

**MARTÍN ROJAS RAMÍREZ**  
**JEFE DE LA DIVISIÓN DE ESTUDIOS PROFESIONALES**



c.p. archivo



Av. Instituto Tecnológico No. 418, San Andrés Ahuashuatepec, Municipio de Tzompantepec,  
Tlaxcala, Mex. C.P. 90491 Tel. (241) 4172010 Ext. 304, e-mail:  
dep\_titulacion@apizaco.tecnm.mx



**2024**  
**Felipe Carrillo**  
**PUERTO**  
Presidencia del Poder Judicial  
del Poder Judicial  
del Poder Judicial

## **Agradecimientos**

Para empezar, quiero agradecer al profesor José Federico Ramírez Cruz quien fue mi asesor en este proyecto, darle las gracias por brindarme un poco de su tiempo, apoyo y conocimiento para poder realizarlo, con su ayuda se pudo terminar el proyecto con un buen resultado, así también ayudándome a progresar profesionalmente durante mi estancia en la universidad.

Igualmente me gustaría agradecer al Instituto Tecnológico de Apizaco por darme la oportunidad de poder estudiar una carrera donde me pude formar académicamente gracias a sus docentes que me brindaron tiempo, pero más importante; me brindaron su conocimiento para poder aprender lo necesario y que me sirviera para poder aplicarlos en mi vida profesional.

Quiero agradecer a mis padres que para mí fueron los más importantes en este proceso porque durante toda mi vida gracias a ellos he tenido la motivación de salir adelante, siempre han sido el motivo por el que hago todo y que gracias a su esfuerzo de todos los días he podido llegar hasta aquí.

También a mis amigos Karina, Luis David, Carmen y David que durante mi proceso académico estuvieron siempre para mí, siempre me apoyaron en cada momento, son las mejores personas que pude conocer en la universidad, siempre me impulsaron para poder llegar a mi meta.

Muchas gracias a todos por apoyarme en este proceso importante en mi vida.

## **Resumen**

El reconocimiento de objetos es una tecnología clave presente en los vehículos sin conductor que les permite reconocer una señal de stop o distinguir entre un peatón y una farola. También resulta útil en diversas aplicaciones, tales como la identificación de enfermedades en las bioimágenes, la inspección industrial y la visión robótica.

La tecnología de reconocimiento de objetos y visión por computadora ahora también está disponible para los experimentadores, con varios kits y cámaras con diversas capacidades disponibles.

Cuando las personas miramos una fotografía o vemos un vídeo, detectamos con rapidez personas, objetos, lugares y detalles visuales., entonces como principal concepto de este proyecto era que con una cámara Pixy poder reconocer los objetos que le mostramos, que la cámara los reconociera y mediante esto crear un pequeño robot el cual pudiera seguir esos objetos en donde sea que estuvieran ubicados a como lo hace una persona.

Como primera instancia, se comprobó que la cámara Pixy funcione de manera correcta y fácil al reconocer los objetos mostrados; más tarde se siguió el mismo procedimiento para conseguir la implementación del Arduino con la cámara Pixy ya que la cámara Pixy puede comunicarse fácilmente con un microcontrolador utilizando sus interfaces de comunicación (UART, SPI, I2C, USB). Se conecta al puerto ICSP de la placa Arduino.

## INDICE

|                                                                                                        |           |
|--------------------------------------------------------------------------------------------------------|-----------|
| <b>Agradecimientos .....</b>                                                                           | <b>2</b>  |
| <b>Resumen.....</b>                                                                                    | <b>3</b>  |
| <b>Introducción .....</b>                                                                              | <b>8</b>  |
| <b>1. Descripción de la empresa u organización y del puesto o área del trabajo del estudiante.....</b> | <b>9</b>  |
| 1.1. Descripción de la institución .....                                                               | 9         |
| 1.2. Misión .....                                                                                      | 9         |
| 1.3. Visión.....                                                                                       | 9         |
| 1.4. Valores .....                                                                                     | 9         |
| 1.5. Ubicación .....                                                                                   | 9         |
| <b>2. Descripción del área de trabajo .....</b>                                                        | <b>10</b> |
| 2.1. Problemas a resolver .....                                                                        | 10        |
| <b>3. Objetivos .....</b>                                                                              | <b>10</b> |
| 3.1. Objetivo general.....                                                                             | 10        |
| 3.2. Objetivos específicos .....                                                                       | 10        |
| 3.3. Justificación .....                                                                               | 11        |
| <b>4. Marco teórico.....</b>                                                                           | <b>11</b> |
| 4.1. Robot Móvil.....                                                                                  | 11        |
| 4.2. Cámara Pixy .....                                                                                 | 12        |
| 4.3. PixyMon.....                                                                                      | 12        |
| 4.4. Arduino .....                                                                                     | 12        |
| 4.5. L298N Controlador de Motores .....                                                                | 12        |
| 4.6. Microcontrolador .....                                                                            | 12        |
| 4.7. Conector SPI .....                                                                                | 12        |
| 4.8. Falsos negativos .....                                                                            | 13        |
| 4.9. Falsos positivos .....                                                                            | 13        |
| 4.10. Conector ICSP .....                                                                              | 13        |
| 4.11. API.....                                                                                         | 13        |
| <b>5. Desarrollo del proyecto.....</b>                                                                 | <b>14</b> |
| <b>6. Procedimiento y descripción de las actividades realizadas .....</b>                              | <b>15</b> |
| 6.1. Cámara Pixy .....                                                                                 | 15        |
| 6.2. Funciones agregadas a Pixy2 .....                                                                 | 15        |

|            |                                                                                              |           |
|------------|----------------------------------------------------------------------------------------------|-----------|
| 6.3.       | 60 fotogramas por segundo .....                                                              | 16        |
| 6.4.       | Dinosaurios morados y (otras cosas) .....                                                    | 16        |
| 6.5.       | Firmas de siete colores .....                                                                | 16        |
| 6.6.       | ¿Cómo conectar la Cámara Pixy?.....                                                          | 17        |
| 6.6.1.     | PixyMon permite ver lo que ve Pixy.....                                                      | 17        |
| 6.7.       | ¿Cómo reconoce Pixy los objetos? .....                                                       | 19        |
| 6.8.       | Varias firmas.....                                                                           | 21        |
| 6.9.       | ¿Cómo enseñarle a Pixy un objeto a través de PixyMon? .....                                  | 22        |
| 6.10.      | Afinación de firma .....                                                                     | 23        |
| 6.11.      | Ajustar el mínimo de fotogramas por segundo.....                                             | 26        |
| 6.12.      | Aumentar el filtrado de bloques .....                                                        | 27        |
| 6.13.      | Activación del resaltado de sobreexposición y ajuste del brillo/exposición de la Cámara..... | 28        |
| <b>7.</b>  | <b>Conectar Pixy2 a Arduino .....</b>                                                        | <b>30</b> |
| 7.1.       | Cable de conexión SPI.....                                                                   | 30        |
| 7.2.       | Biblioteca de Arduino .....                                                                  | 30        |
| 7.3.       | Monitor serie .....                                                                          | 31        |
| 7.4.       | Matriz blocks.....                                                                           | 32        |
| <b>8.</b>  | <b>API de Arduino.....</b>                                                                   | <b>34</b> |
| 8.1.       | Incluir archivos .....                                                                       | 34        |
| 8.2.       | Solución de problemas.....                                                                   | 35        |
| <b>9.</b>  | <b>L298N Controlador de Motores .....</b>                                                    | <b>36</b> |
| 9.1.       | ¿Qué es L298N?.....                                                                          | 36        |
| 9.2.       | Conexión y funcionamiento .....                                                              | 36        |
| 9.3.       | Motor de corriente continua.....                                                             | 38        |
| <b>10.</b> | <b>Resultados .....</b>                                                                      | <b>39</b> |
| 10.1.      | Construcción del robot móvil con Arduino y Cámara Pixy .....                                 | 39        |
| 10.2.      | Cableado del motor L298N a Arduino.....                                                      | 39        |
| 10.3.      | Diagrama de flujo .....                                                                      | 44        |
| 10.4.      | Código en Arduino .....                                                                      | 45        |
| 10.4.1     | Explicación de lo que hace el código.....                                                    | 47        |
| <b>11.</b> | <b>Pruebas .....</b>                                                                         | <b>50</b> |

|       |                                                   |           |
|-------|---------------------------------------------------|-----------|
| 11.1. | imágenes de las pruebas .....                     | 50        |
| 12.   | <b>Conclusiones .....</b>                         | <b>51</b> |
| 12.1  | Conclusiones del proyecto .....                   | 51        |
| 12.2  | Recomendaciones .....                             | 51        |
| 12.3  | Experiencia personal y profesional adquirida..... | 51        |
| 13.   | <b>Competencias desarrolladas .....</b>           | <b>52</b> |
| 14.   | <b>Fuentes de información.....</b>                | <b>53</b> |

## Índice de Ilustraciones

|                 |                                                                                                     |    |
|-----------------|-----------------------------------------------------------------------------------------------------|----|
| Ilustración 1.  | Prototipo del proyecto.....                                                                         | 14 |
| Ilustración 2.  | Estructura de la Cámara Pixy .....                                                                  | 15 |
| Ilustración 3.  | Ejemplo de robot móvil (Imagen de Google).....                                                      | 16 |
| Ilustración 4.  | Versiones de Cámara Pixy .....                                                                      | 17 |
| Ilustración 5.  | Softwares para instalar PixyMon .....                                                               | 17 |
| Ilustración 6.  | Instalando controladores de Pixy .....                                                              | 18 |
| Ilustración 7.  | Objeto de color mostrado en Pixy (Imagen de Google) .....                                           | 18 |
| Ilustración 8.  | Objetos que reconoce y no reconoce Pixy.....                                                        | 19 |
| Ilustración 9.  | Tubo de luz (Imagen de Google) .....                                                                | 20 |
| Ilustración 10. | Objeto reconocido y no reconocido por PixyMon.....                                                  | 20 |
| Ilustración 11. | Enseñando a Pixy un objeto (Imagen de Google).....                                                  | 22 |
| Ilustración 12. | Objeto aprendido por Pixy (Imagen de Google).....                                                   | 22 |
| Ilustración 13. | Pixy rastrea el objeto (Imagen de Google).....                                                      | 23 |
| Ilustración 14. | Panel Tuning (Imagen de Google).....                                                                | 23 |
| Ilustración 15. | Ajustando la inclinación de la firma hacia la izquierda (falsos positivos) (Imagen de Google) ..... | 24 |
| Ilustración 16. | Ajustando la inclinación de la firma hacia la derecha (falsos negativos) (Imagen de Google) .....   | 24 |
| Ilustración 17. | Detección fuerte de la firma (Imagen de Google).....                                                | 25 |
| Ilustración 18. | Ajustando el mínimo de fotogramas por segundo (Imagen de Google).....                               | 26 |
| Ilustración 19. | Fps altos y Fps bajos (Imagen de Google) .....                                                      | 26 |
| Ilustración 20. | Filtrado de bloques (Imagen de Google).....                                                         | 27 |
| Ilustración 21. | Resaltado de sobreexposición (Imagen de Google) .....                                               | 28 |
| Ilustración 22. | Imagen sobreexpuesta (Imagen de Google).....                                                        | 28 |
| Ilustración 23. | Configuración para ajuste de brillo de la Cámara (Imagen de Google).....                            | 29 |
| Ilustración 24. | Imagen que muestra sobreexposición de un objeto (Imagen de Google).....                             | 29 |
| Ilustración 25. | Conexión de la Cámara Pixy con cable SPI (Imagen de Google) ...                                     | 30 |
| Ilustración 26. | Descargar biblioteca para Arduino.....                                                              | 30 |

|                                                                                                   |                 |
|---------------------------------------------------------------------------------------------------|-----------------|
| Ilustración 27. Librería de Arduino .....                                                         | 31              |
| Ilustración 28. Ejemplo (hello world en Arduino) .....                                            | 31              |
| Ilustración 29. Configuración para inicializar monitor .....                                      | 32              |
| Ilustración 30. Salida del monitor .....                                                          | 33              |
| Ilustración 31. Programa color_connected_components detectando objetos<br>(Imagen de Google)..... | 33              |
| Ilustración 32. Error en Arduino en Hello world.....                                              | 35              |
| Ilustración 33. L298N Estructura de controlador de motores (Imagen de Google)                     | 36              |
| Ilustración 34. Conexión de Arduino con L298N (Imagen de Google).....                             | 37              |
| Ilustración 35. Chasis Auto Robot.....                                                            | 39              |
| Ilustración 36. Esquema de conexión L298N a Arduino.....                                          | 40              |
| Ilustración 37. Prototipo de Robot Móvil, vista por arriba .....                                  | 41              |
| Ilustración 38. Prototipo armado con Arduino y L298N, vista de lado .....                         | 41              |
| Ilustración 39. Imágenes del Robot para detectar objetos .....                                    | 42              |
| Ilustración 40. Robot para detectar Objetos.....                                                  | 43              |
| Ilustración 41. Diagrama de flujo del Robot Móvil .....                                           | 44              |
| Ilustración 42. Código en Arduino parte 1.....                                                    | 45              |
| Ilustración 43. Código en Arduino parte 2.....                                                    | 45              |
| Ilustración 44. Código en Arduino parte 3.....                                                    | 46              |
| Ilustración 45. Código en Arduino parte 4.....                                                    | 47              |
| Ilustración 46. Mostrándole el Objeto al Robot Móvil                                              |                 |
| Ilustración 47. Robot Móvil buscando el Objeto .....                                              | 49              |
| Ilustración 48. Deteniéndose cuando llega al objeto                                               | Ilustración 49. |
| Ilustración 49. Llegando al objeto .....                                                          | 49              |
| Ilustración 50. Siguiendo el Objeto -----                                                         | 49              |
| Ilustración 51. Siguiendo el Objeto hacia donde lo mueva.....                                     | 49              |
| Ilustración 52. Deteniéndose cuando llega al objeto -----                                         | 49              |
| Ilustración 53. Llegando al objeto .....                                                          | 49              |

## Introducción

La robótica educativa es una disciplina **concebida para que los estudiantes se inicien desde edades muy tempranas en la robótica y la programación de forma interactiva.**

En el caso de la enseñanza infantil y primaria, la robótica educativa pone a disposición de los alumnos **todo lo necesario para construir y programar de forma sencilla un robot capaz de ejecutar diferentes tareas.** Asimismo, también hay robots más avanzados y de mayor coste destinados a la educación secundaria y superior. En cualquier caso, **la complejidad de la disciplina siempre se adapta a la edad de los alumnos.**

Otro importante beneficio de los robots educativos es que enseñan a los niños a familiarizarse y aprender conceptos básicos de programación, una habilidad que va cobrando mayor relevancia cada año.

Es por eso que se decidió hacer el robot con una Cámara Pixy conectada a un Microcontrolador Arduino que sea Movido por un robot Móvil; y este pueda reconocer figuras geométricas para poder facilitar el aprendizaje de los niños de una forma divertida y llamativa, ya que hoy en día la tecnología se usa para todo y es algo que les llama mucho la atención.

## 1. Descripción de la empresa u organización y del puesto o área del trabajo del estudiante

### 1.1. Descripción de la institución

Personalidades Emilio Sánchez Piedras, entonces Gobernador Constitucional, comprometido con la juventud del estado, contando con el apoyo decisivo del Presidente de la República, el Lic. Luis Echeverría Álvarez, personaje visionario del futuro de la educación y de la juventud de esta región del país, gestiona y funda el Instituto Tecnológico Regional de Apizaco e inicia sus actividades el 1 de octubre del año de 1975.

El Tecnológico de Apizaco, desde hace más de cuatro décadas, se ha dedicado a propiciar alternativas reales y sustentables para la juventud de la región. Es una Institución que pertenece al Tecnológico Nacional de México, del cual formamos parte de 254 Tecnológicos distribuidos en los 32 estados del país.

Los planes de estudio están basados en una estructura genérica, una especialidad y una residencia profesional.

### 1.2. Misión

Formar profesionistas que sean líderes visionarios, emprendedores, con capacidad de generar soluciones disruptivas, innovando en la ciencia y la tecnología, comprometidos con el desarrollo económico, responsabilidad social teniendo una visión sostenible y sustentable.

### 1.3. Visión

Ser una Institución de educación superior tecnológica de vanguardia, formadora de ciudadanos del mundo, con conocimientos intercultural, respetando la diversidad social.

### 1.4. Valores

- **Honestidad:** Se demuestra día a día a través de la transparencia con la ejecución de las actividades laborales cotidianas.
- **Responsabilidad:** Es la actitud que caracteriza al personal en el cumplimiento de su tarea con eficacia y eficiencia en beneficio de la comunidad.
- **Lealtad:** Es la actitud que caracteriza al personal del Instituto Tecnológico de Apizaco, aplica sus conocimientos y habilidades en el logro de las metas institucionales, sin olvidar en ningún momento el lema:

*“Pensar para servir, servir para triunfar”*

### 1.5. Ubicación

Carretera Apizaco Tzompantepec, esquina con Av. Instituto Tecnológico S/N, Conurbado Apizaco – Tzompantepec, Tlaxcala, Mex. C.P. 90491

## **2. Descripción del área de trabajo**

El instituto Tecnológico de Apizaco ofrece una ingeniería en Tecnologías de la Información y Comunicación con el fin de ofrecer a sus estudiantes una mejor preparación, aprendizaje y enseñanza por parte de la institución y los profesores que la conforman. En este caso desarrollar un proyecto dentro de la institución a cargo de un asesor, con el objetivo de brindarnos sus conocimientos en el proyecto para una mejor realización del mismo y preparándonos para un entorno laboral.

### **2.1. Problemas a resolver**

La robótica en la educación se ha venido implementando en diferentes países de Europa y América, entre otros, haciendo cada vez más popular el uso de la robótica educativa dentro y fuera de los planes curriculares de diferentes centros educativos alrededor del mundo.

Se han desarrollado diversas iniciativas que buscan preparar a los profesores de educación básica para incorporar la robótica educativa como una estrategia para la enseñanza de las matemáticas; en este caso es crear un robot móvil con el fin de fomentar la innovación autónoma y el pensamiento lógico de los niños desde edades tempranas.

De forma lúdica los niños aprenden ciencia, matemáticas, resolución de problemas y creatividad.

## **3. Objetivos**

### **3.1. Objetivo general**

Realizar el prototipo para reconocer objetos utilizando una Cámara Pixy y provocar que el robot se mueva hacia el objeto programado.

### **3.2. Objetivos específicos**

- Diseñar un programa en Arduino y/o Python para interactuar con la cámara PixyCam
- Interactuar con el módulo de control de motores de las llantas para indicarle el movimiento
- Lograr la conexión de la cámara Pixy con el Arduino.
- Aprender el lenguaje de programación Arduino.
- Hacer que la cámara Pixy reconozca los objetos mostrados delante de ella.
- Conectar el Arduino a los motores para que estos sigan a los objetos ya reconocidos por Pixy.

### 3.3. Justificación

La robótica educativa ha incursionado en la enseñanza-aprendizaje para motivar a los pequeños aprender. En este sentido se ha propuesto un proyecto para implementar la primera fase del prototipo robot inteligente que enseñe a los niños algunos temas educativos; sin embargo, se necesitan programas que apoyen al robot a moverse utilizando visión por computadora. Este proyecto se propone realizar la parte del programa que guíe al robot hacia los objetos utilizando una cámara.

## 4. Marco teórico

Para el desarrollo de este proyecto se necesitaron herramientas que ya conociáramos o utilizamos a lo largo del aprendizaje brindado. En este caso se utilizó una Cámara Pixy y Arduino para poder hacer la realización de este proyecto, empezando con la comprensión de la Cámara y siguiendo la cronología de pasos para comenzar con el reconocimiento de objetos.

### 4.1. Robot Móvil

Son dispositivos autónomos asistidos por software de robótica e inteligencia artificial, dotados de sensores y de un plano digital del almacén donde desarrollan su gestión.

A su vez se articulan con programas informáticos de control y de gestión de almacenes que planifican las rutas que debe cubrir el dispositivo internamente y con herramientas informáticas de planificación de rutas y monitoreo en tiempo real para los recorridos de última milla.

**La robótica educativa o robótica pedagógica;** es una disciplina concebida para que los estudiantes se inicien desde edades muy tempranas en la robótica y la programación de forma interactiva.

En el caso de la enseñanza infantil y primaria, la robótica educativa pone a disposición de los alumnos todo lo necesario para construir y programar de forma sencilla un robot capaz de ejecutar diferentes tareas. **(Iberdrola, 2022)**

#### 4.2. Cámara Pixy

La Cámara es el sensor principal del robot transportador porque cumple con la función de seguimiento del objeto que la persona portara. Se encuentra ubicada en la parte frontal y mitad del robot para mayor facilidad de detección por parte del sistema, está se comunica con el Arduino a través de protocolo SPI enviando los datos correspondientes que se necesita para cumplir con la programación. **(PIXY, 2021)**

#### 4.3. PixyMon

PixyMon es una aplicación que permite configurar la Cámara y poder ver lo que estamos observando. Se puede ejecutar en varias plataformas diferentes, incluyendo Windows, MacOS y Linux, así como otros sistemas integrados más pequeños como Raspberry Pi y BeagleBone Black. **(Library, 2022)**

#### 4.4. Arduino

Arduino es una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma permite crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores puede darles diferentes tipos de uso. **(Yubal, 2020)**

#### 4.5. L298N Controlador de Motores

El L293D está construido con partes de puentes-H, como ya se había mencionado. Está característica nos permite controlar distintas cantidades de motores. Por ejemplo, puede controlar 4 motores de DC unidireccionalmente o 2 motores-dc en ambas direcciones. **(HETPRO, 2021)**

#### 4.6. Microcontrolador

El Microcontrolador es un circuito integrado que es el componente principal de una aplicación embebida. Es como una pequeña computadora que incluye sistemas para controlar elementos de entrada/salida. También incluye a un procesador y por supuesto memoria que puede guardar el programa y sus variables (flash y RAM). Funciona como una mini PC. Su función es la de automatizar procesos y procesar información. **(Rubén, 2017)**

#### 4.7. Conector SPI

En el bus SPI la comunicación de datos entre maestros y esclavo se realiza en dos líneas independientes, una del maestro a los esclavos, y otra de los esclavos al maestro. Por tanto, **la comunicación es Full Dúplex**, es decir, el maestro puede enviar y recibir datos simultáneamente. **(Luis, 2016)**

#### 4.8. Falsos negativos

Los falsos negativos son cuando Pixy2 no puede detectar un objeto que pretendía detectar con una firma particular. **(PIXY, 2018)**

#### 4.9. Falsos positivos

Los falsos positivos son cuando Pixy2 cree que un objeto coincide con una firma en particular, pero no es el objeto que desea detectar. **(PIXY, 2018)**

#### 4.10. Conector ICSP

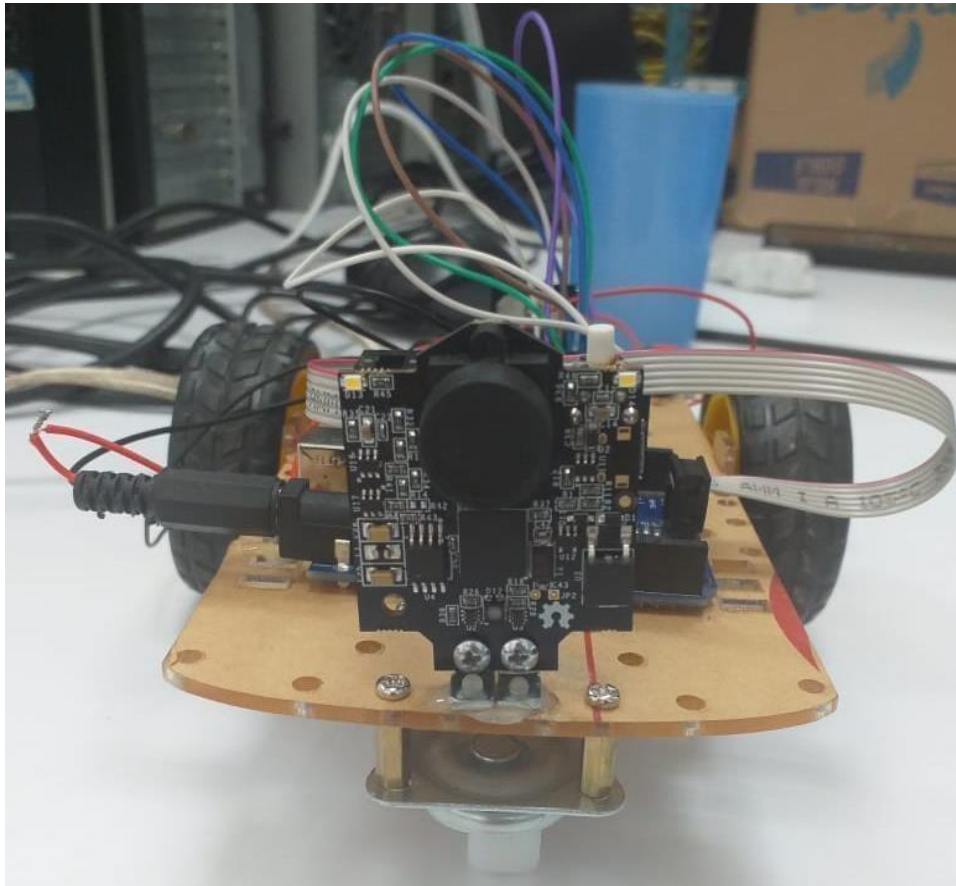
La placa Arduino posee una entrada ICSP (In Chip Serial Programmer) que tiene acceso a la memoria de programa del AVR (Flash), esto es, que puede grabar directamente desde el PC al microcontrolador cualquier programa sin usar el puerto USB. Uno de ellos, el mismo Bootloader de Arduino. **(Wordpress, 2017)**

#### 4.11. API

Una API es una herramienta que puede ser utilizada por otro software de manera independiente. Esta herramienta se presenta, normalmente, en forma de biblioteca y contiene un conjunto de subrutinas, funciones, procedimientos y métodos a los que podemos invocar, dotando a nuestra aplicación de funcionalidades extra.

Arduino tiene su propia API que expone funciones de subrutinas ya programadas y que nos facilitarán el trabajo a la hora de desarrollar nuestro proyecto. **(Luis, 2021)**

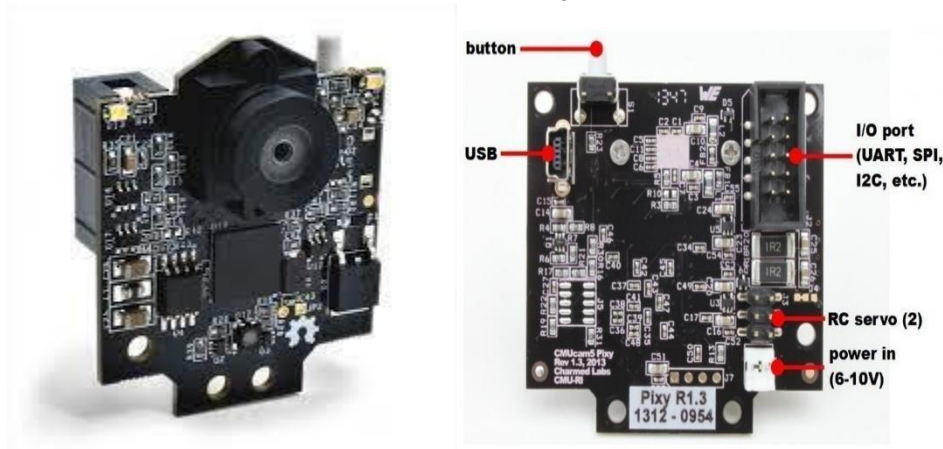
# 5. Desarrollo del proyecto



*Ilustración 1. Prototipo del proyecto*

## 6. Procedimiento y descripción de las actividades realizadas

### 6.1. Cámara Pixy



*Ilustración 2. Estructura de la Cámara Pixy*

Es un sensor de imagen con un potente procesador que se puede programar para enviar sólo la información que estás buscando, evitando sobrecargar tu microcontrolador con datos. Es compatible con Arduino y Raspberry Pi.

Este modelo de Cámara Pixy puede aprender a detectar objetos que le enseñe, simplemente presionando un botón. Además, tiene nuevos algoritmos que detectan y rastrean líneas para su uso con robots de seguimiento de líneas. Los nuevos algoritmos también pueden detectar intersecciones y "señales de tráfico". Las señales de tráfico pueden indicarle a su robot qué hacer, como girar a la izquierda, girar a la derecha, reducir la velocidad, etc.

### 6.2. Funciones agregadas a Pixy 2.1

- Detecta líneas, intersecciones y códigos de barras pequeños, diseñados para robots de seguimiento de líneas
- Velocidad de fotogramas mejorada: 60 fotogramas por segundo
- Se han agregado algoritmos de seguimiento a la detección de objetos basada en colores
- Bibliotecas mejoradas y simplificadas para Arduino, Raspberry Pi y otros controladores
- Fuente de luz integrada

### 6.3. 60 fotogramas por segundo

#### ¿Qué significa "60 fotogramas por segundo"?

En resumen, significa que Pixy es rápido; ya que procesa un cuadro de imagen completo cada 1/60 de segundo (16,7 milisegundos); esto significa que obtiene una actualización completa de todas las posiciones de los objetos detectados cada 16,7 ms. A este ritmo, es posible rastrear la trayectoria de la pelota que cae/rebota.

### 6.4. Dinosaurios morados y (otras cosas)

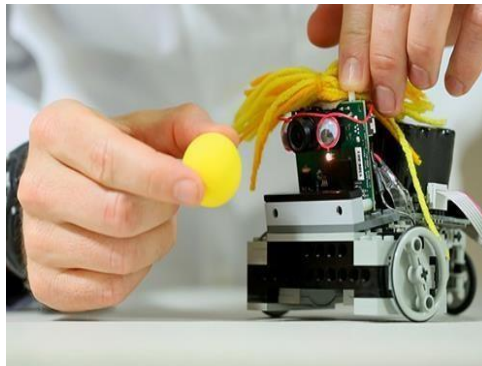
Utiliza un algoritmo de filtrado basado en colores para detectar objetos llamado algoritmo **Color Connected Components (CCC) o Componentes Conectados por Color**. Los métodos de filtrado basados en colores son populares porque son rápidos, eficientes y relativamente robustos.

La mayoría de nosotros estamos familiarizados con RGB (rojo, verde y azul) para representar colores. Pixy calcula el color (tono) y la saturación de cada píxel RGB del sensor de imagen y los utiliza como parámetros de filtrado primarios.

El tono de un objeto permanece prácticamente sin cambios con los cambios en la iluminación y la exposición. Los cambios en la iluminación y la exposición pueden tener un efecto frustrante en los algoritmos de filtrado de color, provocando que se rompan. El algoritmo de filtrado es robusto cuando se trata de cambios de iluminación y exposición.

### 6.5. Firmas de siete colores

El algoritmo CCC de este modelo recuerda hasta 7 firmas de color diferentes, lo que significa que, si tiene 7 objetos diferentes con colores únicos, el algoritmo de filtrado de color no tendrá problemas para identificarlos. Si necesita más de siete, puede utilizar códigos de color.



*Ilustración 3. Ejemplo de robot móvil (Imagen de Google)*

## 6.6. ¿Cómo conectar la Cámara Pixy?

**Paso 1.** Descargar el Programa PixyMon que este nos permitirá configurar la Cámara y ver lo que ve.

### 6.6.1. PixyMon permite ver lo que ve Pixy

PixyMon es una aplicación que se ejecuta en Windows, MacOS y Linux, permite ver lo que ve la cámara, ya sea como video sin procesar o procesado. También permite configurar Pixy, establecer el puerto de salida y administrar firmas de color. PixyMon se comunica con la cámara a través de un cable mini USB estándar.

En la página web de Pixy seleccionamos la cámara que tenemos; que en este caso es la cámara Pixy 2.1.

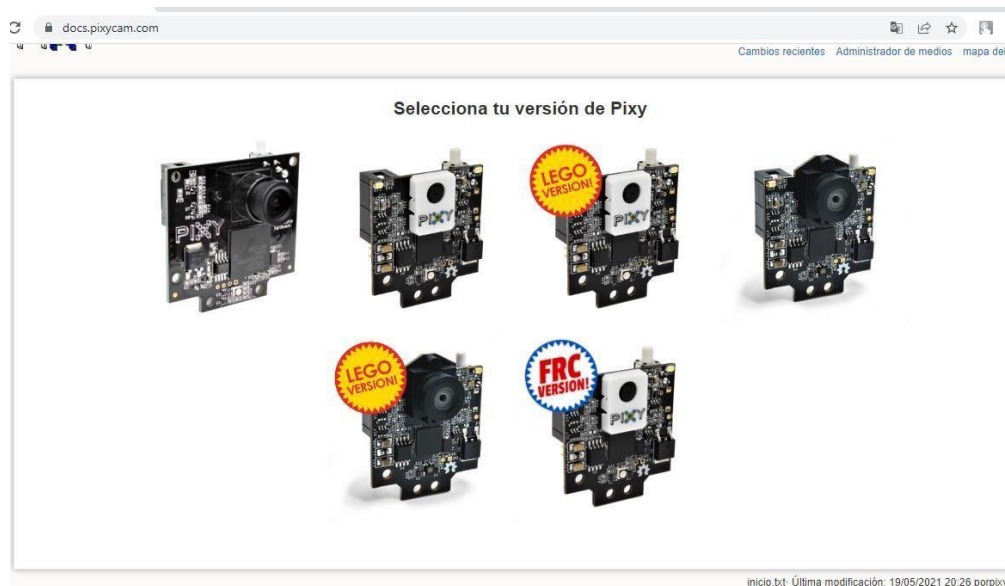


Ilustración 4. Versiones de Cámara Pixy

**Paso 2.** Nos manda a la documentación de la cámara, todo lo que se puede descargar para la cámara y en este caso vamos a instalar el PixyMon; en este caso para Windows.

### Instalando PixyMon v2 para...

- [Mac](#)
- [Windows Vista, 7, 8 o 10](#)
- [Windows XP](#)
- [linux](#)

Ilustración 5. Softwares para instalar PixyMon

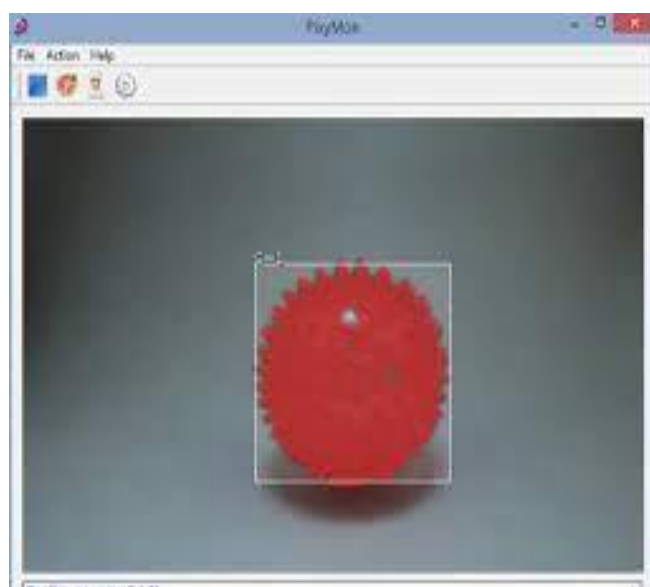
Al ejecutar el instalador de PixyMon; se abrirá una ventana preguntándonos si queremos instalar los controladores para nuestro dispositivo, controladores que vienen con el instalador PixyMon y que son necesarios de instalar para poder conectar con la cámara.



*Ilustración 6. Instalando controladores de Pixy*

**Paso 3.** Después de haberse instalado los controladores correctamente e instalado PixyMon, procedemos a conectar primero el cable USB que trae la cámara a nuestra computadora y después ejecutamos el programa PixyMon.

Al abrir PixyMon, este se conectará y comenzará a mostrar los datos de los componentes conectados en color, en este caso objetos de color que le vamos a ir mostrando para que los pueda reconocer.



*Ilustración 7. Objeto de color mostrado en Pixy (Imagen de Google)*

### 6.7. ¿Cómo reconoce Pixy los objetos?

Utiliza un algoritmo de filtrado de color basado en tonos para detectar objetos. Dado que usa tono (color), el objeto debe tener un tono distinto.

Es por eso que no reconoce objetos de color negro, blanco y gris ya que estos no tienen tono, deben ser objetos que resalten su color.



*Ilustración 8. Objetos que reconoce y no reconoce Pixy*

Teniendo en cuenta que reconoce y que no reconoce Pixy; vamos a mostrarle objetos a la Cámara.

**Paso 1.** Ya que tenemos conectado; la Cámara pasará por una serie de destellos de LED. Esperamos a que el LED se apague antes de enseñarle un objeto.

Ahora, mantenemos presionado el botón en la parte superior de la Cámara y después de aproximadamente 1 segundo, el LED se encenderá, primero blanco, luego rojo, luego otros colores, pero cuando este se ponga rojo, soltamos el botón.

**Paso 2.** Utiliza un algoritmo de crecimiento de regiones para tratar de determinar qué píxeles son parte de su objeto y qué píxeles son parte del fondo. Usando estos píxeles, intentará crear un modelo estadístico de su objeto para que pueda detectarlo de manera confiable bajo diferentes condiciones de iluminación. Use el color del LED como retroalimentación para determinar si tiene un buen bloqueo en el objeto y use las siguientes pautas para juzgar:

- Cuando Pixy se ha bloqueado en el objeto, el color del LED debe coincidir con el color de nuestro objeto.
- Cuanto más brillante sea el LED, mejor será la cerradura.

- La cuadrícula de crecimiento de regiones en la ventana de PixyMon se ve cómo el tamaño de la cuadrícula corresponde con el color y el brillo del LED. Una cuadrícula que cubre más del objeto es mejor que una cuadrícula que cubre solo una parte del objeto y da como resultado un LED más brillante.

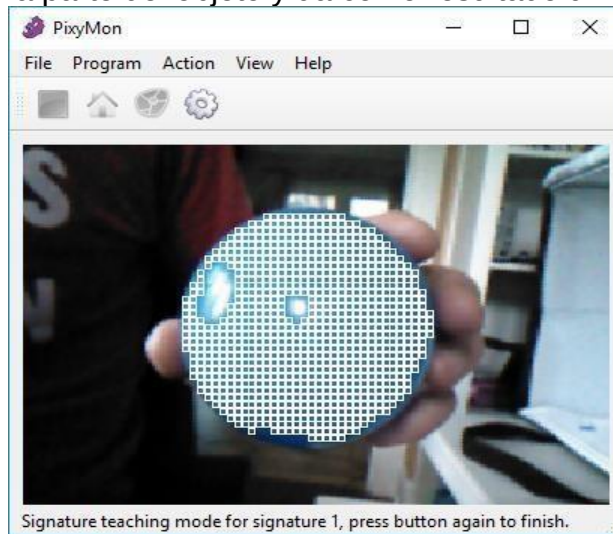


Ilustración 9. Tubo de luz (Imagen de Google)

**Paso 3.** Cuando soltamos el botón, entrará en lo que se llama modo de "**tubo de luz**", donde el color del LED es el color del objeto que la Cámara ha "fijado".

Al mostrarle un objeto de color; el color del LED será el color del objeto que le estamos mostrando y al reconocer el color, apretaremos el botón y lo soltaremos y el Led parpadeara un par de veces para indicar que PixyMon ya ha aprendido nuestro objeto y así comenzara a rastrear el objeto.

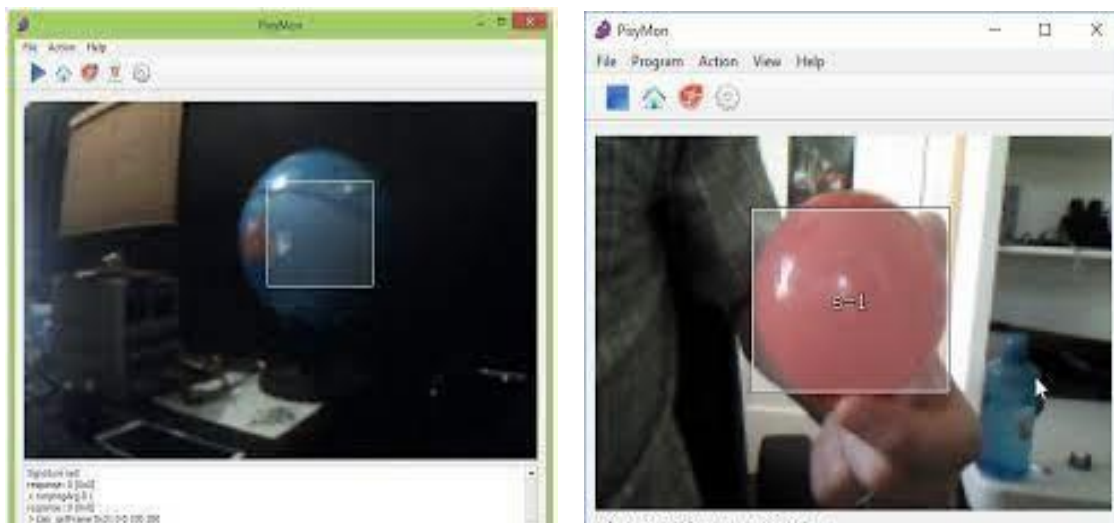


Ilustración 10. Objeto reconocido y no reconocido por PixyMon

## 6.8. Varias firmas

La Cámara puede aprender hasta siete firmas de color. Podemos enseñarle la firma del primer color soltando el botón cuando el LED se vuelve rojo. Si seguimos manteniendo pulsado el botón, el LED se pondrá naranja, amarillo, etc., indicando las firmas de color restantes.

Aquí están las firmas en orden:

1. Rojo
2. Naranja
3. Amarillo
4. Verde
5. Cian (azul claro)
6. Azul
7. Violeta

El número de firma de color se determina cuando soltamos el botón. Soltamos el botón cuando el LED esté amarillo y esté enseñando la firma 3. Soltamos el botón cuando el LED esté azul y esté enseñando la firma 6. Estos colores **no están relacionados con el tono del objeto**. Los colores se utilizan únicamente para indicar el número de firma.

Entonces, por ejemplo, la firma 1 puede ser un objeto amarillo, aunque la firma 1 se indique con un LED rojo, y la firma 2 podría ser un objeto rosa, aunque la firma 2 se indique con un LED naranja.

Después de enseñarle una firma, guarda la firma en flash, de modo que cuando encienda y encienda su Cámara, recordará las firmas que le enseñó y seguirá rastreando los objetos que coincidan con estas firmas.

### 6.9. ¿Cómo enseñarle a Pixy un objeto a través de PixyMon?

Esto puede ser útil si el objeto que deseamos enseñar es pequeño o si deseamos tener más control sobre qué píxeles se utilizan para la enseñanza.

Comenzamos conectando el cable USB entre la Cámara y la computadora y ejecutando PixyMon.

Ahora sostenemos el objeto que deseamos enseñar frente a la Cámara y seleccionamos **Acción→Establecer firma 1** en el menú desplegable.

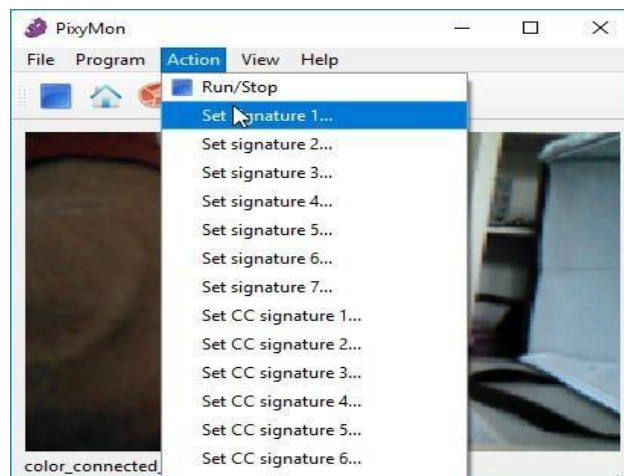


Ilustración 11. Enseñando a Pixy un objeto (Imagen de Google)

Usando el mouse, hacemos clic y arrastramos para seleccionar la región que deseamos que la cámara use para aprender el objeto.

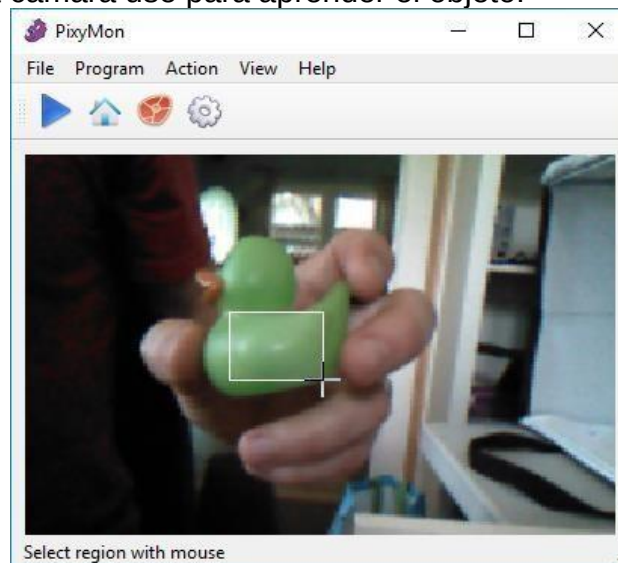


Ilustración 12. Objeto aprendido por Pixy (Imagen de Google)

Después de seleccionar la región, la Cámara "aprenderá" el objeto y automáticamente comenzará a rastrearlo.

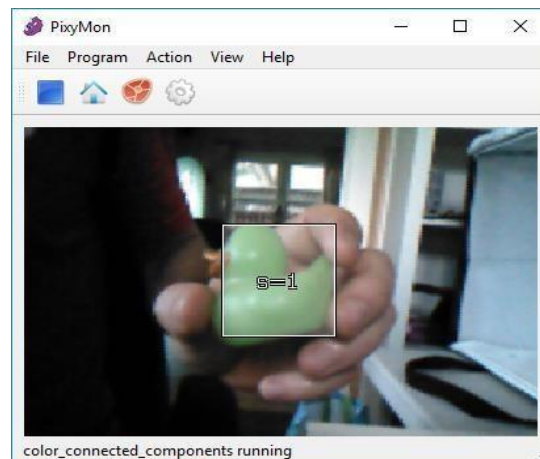


Ilustración 13. Pixy rastrea el objeto (Imagen de Google)

#### 6.10. Afinación de firma

A veces, las firmas de color que le enseñamos necesitan ser "ajustadas", es decir, obtiene algunos falsos positivos (Pixy está detectando objetos que no son los objetos que pretendíamos) o falsos negativos (Pixy no está detectando el objeto que deseamos). simplemente lo enseñó, o está detectando el objeto de forma intermitente).

Podemos modificar las cosas abriendo el cuadro de diálogo Configurar (haciendo clic en el icono de engranaje o seleccionando **Archivo→Configurar**).

Seleccionamos el panel **Tuning en Pixy2 Parameters**.

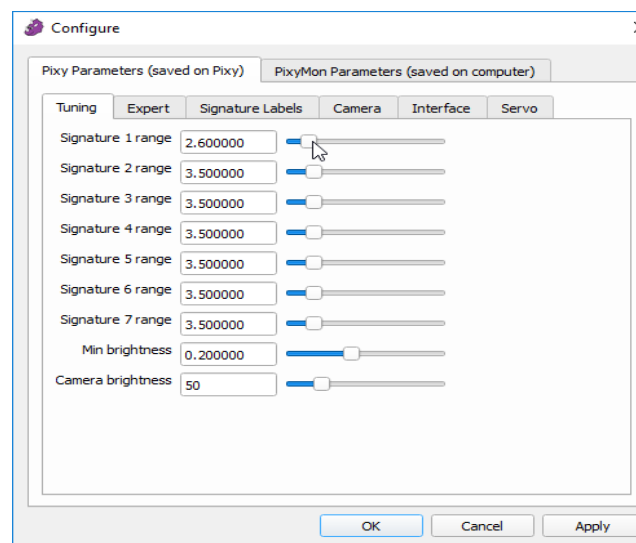
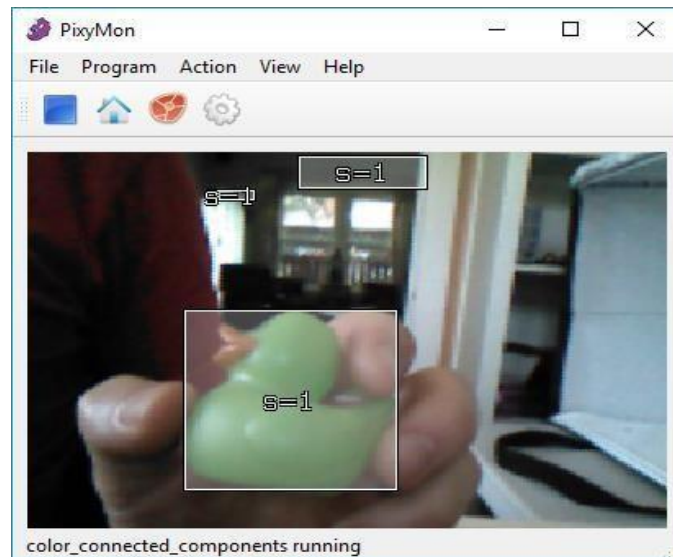


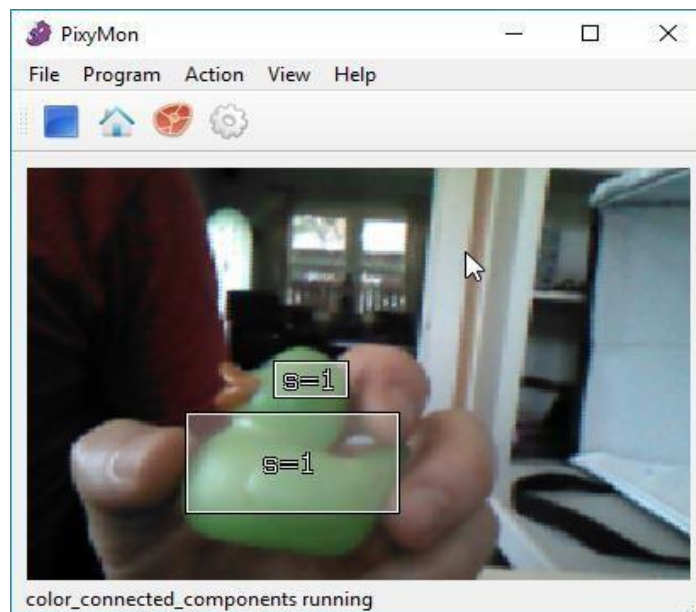
Ilustración 14. Panel Tuning (Imagen de Google)

Usamos el control para deslizarlo hacia el **rango de Firma 1** para ajustar la inclusión de la firma 1 (suponiendo que es la firma 1 lo que deseamos ajustar). Deslizamos hacia la izquierda si queremos ser menos inclusivo (es decir, si vemos falsos positivos, como la imagen a continuación):



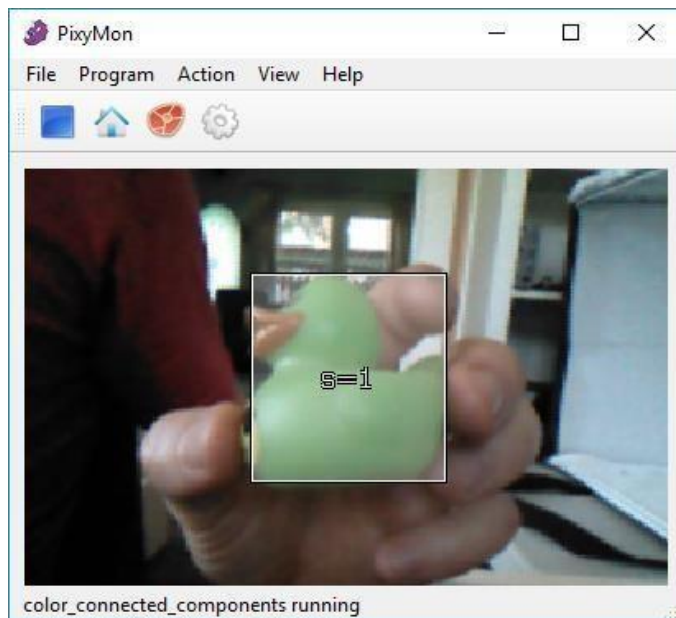
*Ilustración 15. Ajustando la inclinación de la firma hacia la izquierda (falsos positivos) (Imagen de Google)*

o lo deslizamos hacia la derecha si se desea ser más inclusivo (es decir, se está viendo falsos negativos) o si la detección es intermitente o escasa (como en la imagen a continuación):



*Ilustración 16. Ajustando la inclinación de la firma hacia la derecha (falsos negativos) (Imagen de Google)*

Elegimos un valor del control deslizante que proporcione una buena detección fuerte, como la imagen a continuación:



*Ilustración 17. Detección fuerte de la firma (Imagen de Google)*

Podemos ajustar las siete firmas de color de esta manera para maximizar la precisión de detección para todas las firmas. Nos tenemos que asegurar de presionar **Aplicar** o **Aceptar** para guardar los rangos del control deslizante. Los valores ajustados no se guardarán si presionamos **Cancelar** o cerramos el cuadro de diálogo.

### 6.11. Ajustar el mínimo de fotogramas por segundo

Ajustar el control deslizante Mínimo de fotogramas por segundo en la pestaña Cámara a valores más pequeños puede mejorar la calidad de la imagen al reducir el ruido y/o mejorar la exposición en entornos con poca luz. La desventaja de reducir los fotogramas por segundo es que reduce la tasa de actualización.

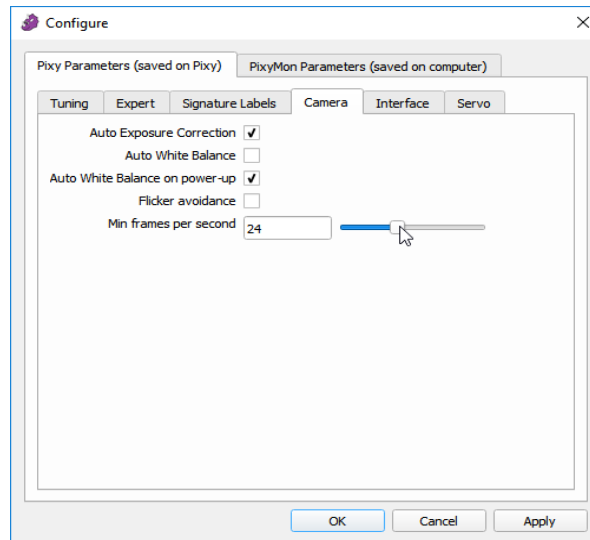


Ilustración 18. Ajustando el mínimo de fotogramas por segundo (Imagen de Google)

Las dos imágenes a continuación muestran una escena con poca luz y la reducción del ruido al reducir los fotogramas por segundo. (El marco izquierdo tiene fps más altos, más ruido).

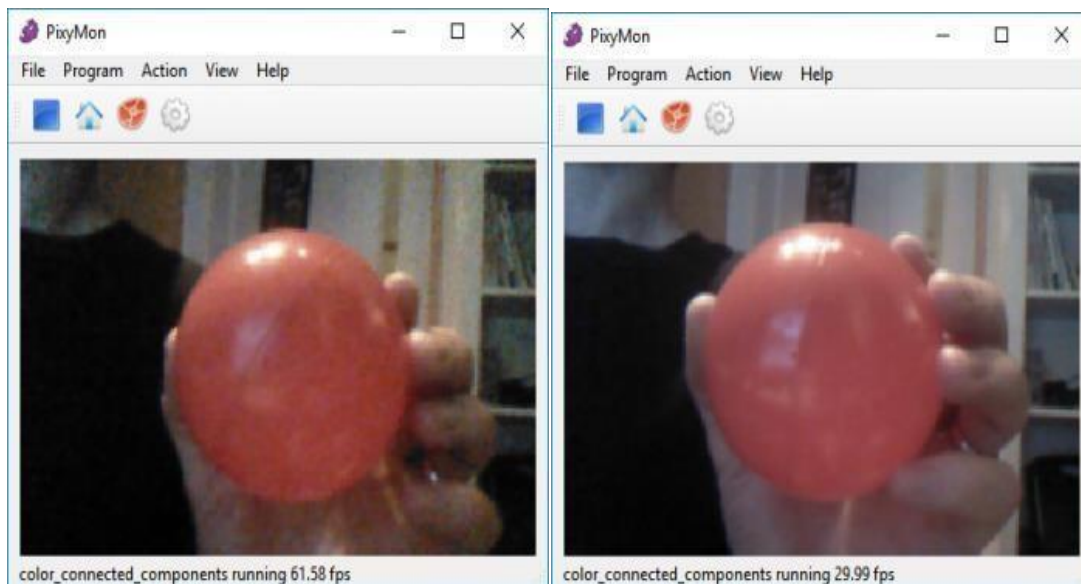


Ilustración 19. Fps altos y Fps bajos (Imagen de Google)

### 6.12. Aumentar el filtrado de bloques

La Cámara Utiliza el filtrado para determinar si es probable que un bloque determinado sea válido o solo una detección de falso positivo. Aumentar el **filtrado de bloques** en la pestaña **Experto** puede reducir la cantidad de detecciones de falsos positivos. El costo de aumentar el filtrado de bloques es una mayor latencia al detectar nuevos objetos. Es decir, tardará más en informar la detección de un nuevo objeto y agregarlo a la lista de objetos rastreados. Una vez que se agrega un objeto a la lista de objetos rastreados, la posición de cada objeto no se filtra para que la dinámica de la posición/movimientos informados del objeto no se vea afectada.

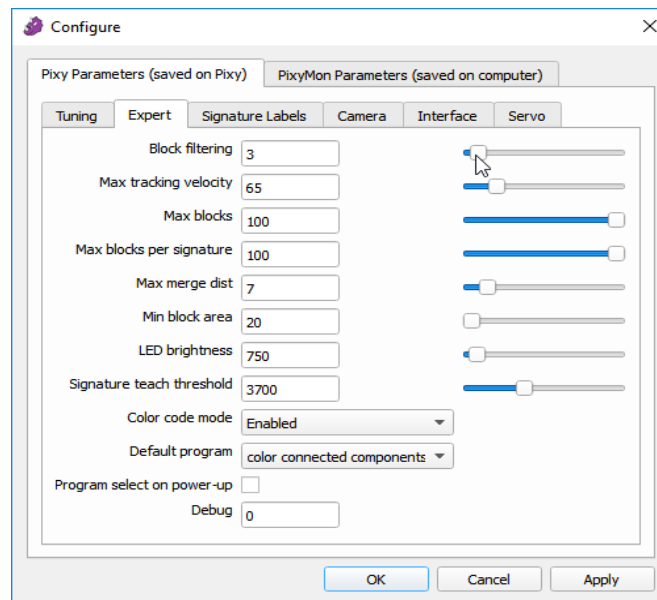


Ilustración 20. Filtrado de bloques (Imagen de Google)

### 6.13. Activación del resaltado de sobreexposición y ajuste del brillo/exposición de la Cámara

Al igual que una cámara, necesita obtener la configuración de exposición correcta o sus imágenes no contendrán suficiente rango dinámico y la precisión de detección se verá afectada. PixyMon tiene una función que resalta las regiones de la imagen que están sobreexpuestas. Podemos habilitarlo haciendo clic en el cuadro junto a **Resaltar sobreexposición** en la **pestaña Parámetros de PixyMon**

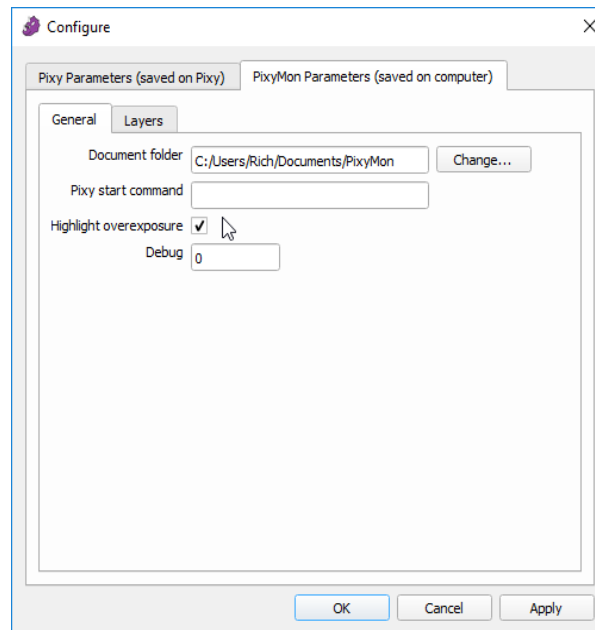


Ilustración 21. Resaltado de sobreexposición (Imagen de Google)

Cuando está habilitado, las partes sobreexpuestas de la imagen se resaltarán en negro, como se muestra a continuación:

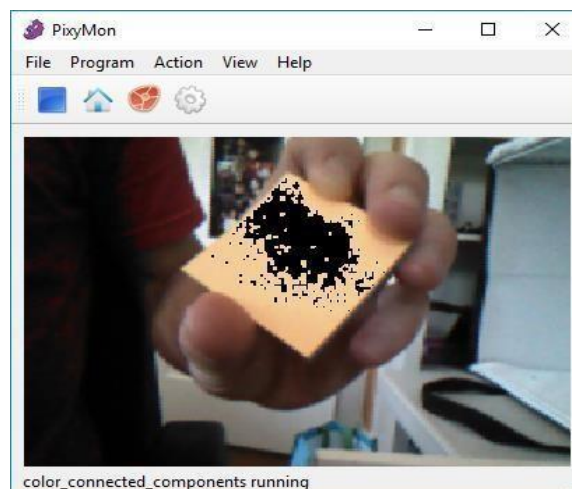


Ilustración 22. Imagen sobreexpuesta (Imagen de Google)

Se recomienda habilitar el resaltado de sobreexposición, porque si el objeto que desea detectar está sobreexpuesto, la precisión de detección se verá afectada. Entonces, después de habilitar el resaltado de sobreexposición, en la pestaña **Ajuste en Parámetros de Pixy** y elegimos una buena configuración para el **Brillo** de la cámara ajustando el control deslizante.

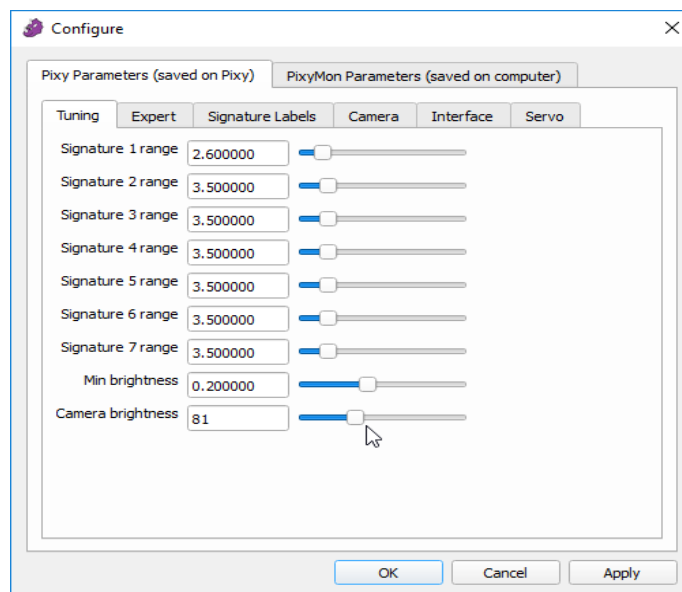


Ilustración 23. Configuración para ajuste de brillo de la Cámara (Imagen de Google)

Una buena configuración de exposición deja casi todos los píxeles del objeto correctamente expuestos, pero no demasiado oscuros (subexpuestos). Está bien que partes pequeñas de nuestro objeto estén sobreexpuestas, especialmente si el objeto es brillante. En la imagen a continuación: se tiene en cuenta que una pequeña parte del objeto tiene una región negra que indica sobreexposición. Esto está bien.



Ilustración 24. Imagen que muestra sobreexposición de un objeto (Imagen de Google)

## 7. Conectar Pixy a Arduino

El puerto USB no es una buena opción para conectar la Cámara a Arduino ya que USB requiere demasiada sobrecarga de memoria para ser práctico y el uso del puerto USB le impediría usar simultáneamente PixyMon como herramienta de depuración.

### 7.1. Cable de conexión SPI

El bus Arduino SPI tiene una frecuencia de reloj de 2 MHz, lo que le permite transferir datos a 2 Mbits/segundo. Con esta conexión, se pueden recuperar datos de la cámara y también se puede controlar los dos servomotores utilizados en el soporte de giro e inclinación opcional.



Ilustración 25. Conexión de la Cámara Pixy con cable SPI (Imagen de Google)

### 7.2. Biblioteca de Arduino

En la página principal de Pixy se descarga la última versión de biblioteca de Arduino con ejemplos.

← → ↻ pixycam.com/downloads-pixy2/

**PIXY** PRODUCTOS PROYECTOS VÍDEOS COMPR

◦ documentos de instalación

## Bibliotecas y ejemplos de Arduino

Las bibliotecas de Arduino permiten que sus bocetos/programas de Arduino hablen con Pixy2.

- [Biblioteca Arduino Pixy2 versión 1.0.3 \(zip\)](#)
  - documentos de instalación

## libpixyusb2

Libpixyusb2 es una biblioteca C/C++ que permite que su controlador basado en Linux (por ejemplo Pixy2) a través de USB.

- La última versión está disponible a través de [github](#)
  - documentos de instalación

Ilustración 26. Descargar biblioteca para Arduino

### 7.3. Monitor serie

Abrimos el IDE de Arduino e importamos la biblioteca Pixy seleccionando **Sketch→Include Library→Add .ZIP Library...** en el IDE de Arduino y luego navegamos hasta el archivo zip de Arduino que acabamos de descargar.

A continuación, cargamos el ejemplo “ccc\_hello\_world” seleccionándolo en **File→Examples→Pixy2**.

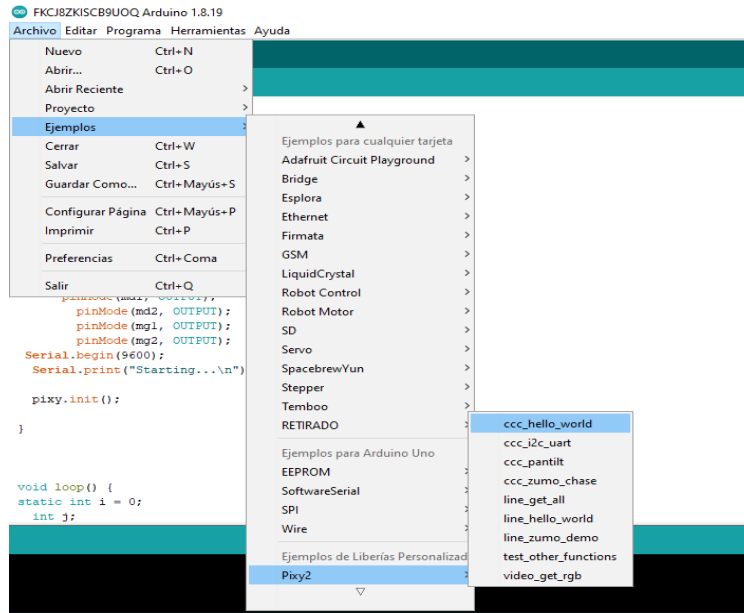


Ilustración 27. Librería de Arduino



Ilustración 28. Ejemplo (hello world en Arduino)

En la sección de configuración se inicializa el monitor serie. Hay que tener en cuenta que está configurado para ejecutarse en 115200 Bps en lugar de los 9600 Bps predeterminados. Debemos asegurarnos de configurar el monitor serial para esta velocidad, la velocidad más alta se utiliza para acomodar la alta velocidad de transferencia de datos de la cámara al Arduino.

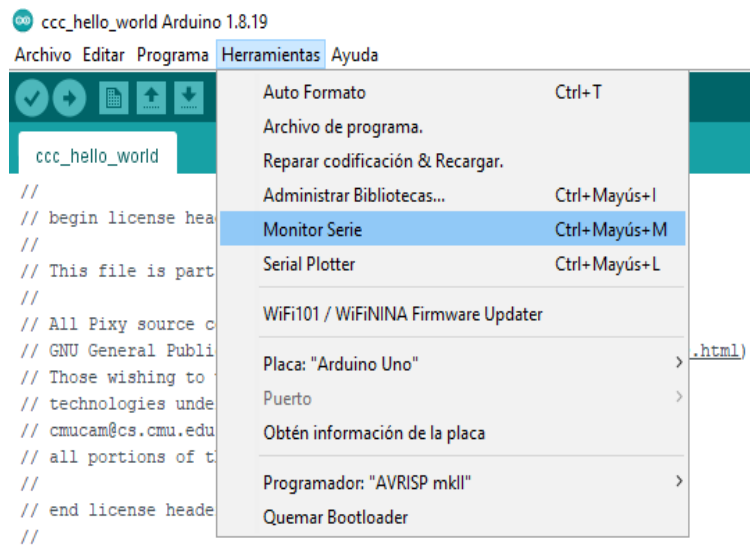


Ilustración 29. Configuración para inicializar monitor

#### 7.4. Matriz blocks

Salida del monitor serie, contenido de la matriz blocks [].

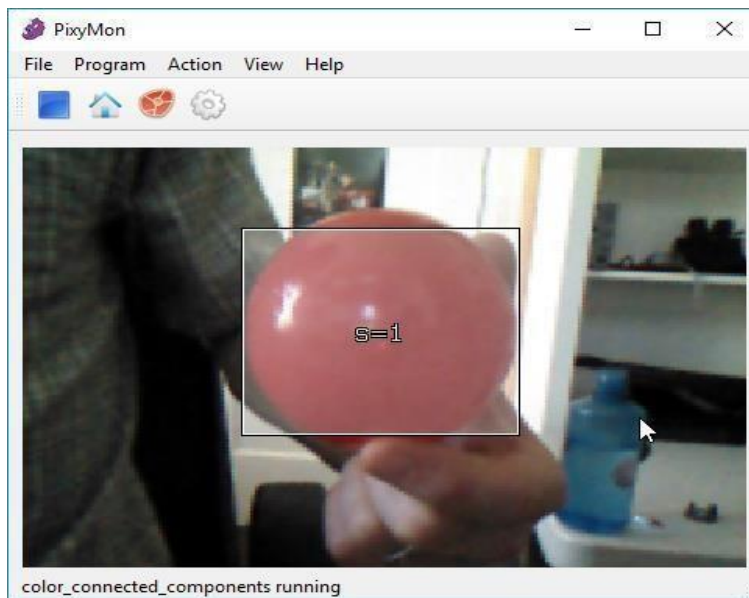
La matriz tiene varios elementos:

- ✚ **Bloque:** Esta es la posición del bloque en la matriz, comenzando en cero.
- ✚ **Sig:** El número de firma, del 1 al 7. No, que se devolverá un número incluso si obtuviste las firmas individuales usando PixyMon.
- ✚ **X e Y:** Las coordenadas del bloque dentro del marco visible.
- ✚ **Ancho:** el ancho del bloque.
- ✚ **Altura:** La altura del bloque.
- ✚ **Índice:** Un número único asignado a un objeto en la pantalla. Esto le permite rastrear objetos incluso cuando hay una multitud de objetos idénticos.
- ✚ **Edad:** Un número del 1 al 255. La "edad" es el número de fotogramas en los que se ha visto este bloque en particular, esto le permite formar una "historia" del objeto.

```
Detectado 1
  bloque 0: sig: 1 x: 160 y: 111 ancho: 140 alto: 135 índice: 38 edad: 42
Detectado 1
  bloque 0: sig: 1 x: 160 y: 112 ancho: 140 alto: 136 índice: 38 edad: 43
Detectado 1
  bloque 0: sig: 1 x: 160 y: 111 ancho: 140 alto: 136 índice: 38 edad: 44
Detectado 1
  bloque 0: sig: 1 x: 160 y: 111 ancho: 140 alto: 135 índice: 38 edad: 45
Detectado 1
  bloque 0: sig: 1 x: 162 y: 109 ancho: 140 alto: 133 índice: 38 edad: 46
Detectado 1
  bloque 0: sig: 1 x: 162 y: 109 ancho: 140 alto: 135 índice: 38 edad: 47
...
```

*Ilustración 30. Salida del monitor*

Solo se imprimirá mensajes si se está ejecutando el programa "color\_connected\_components" y se ve un objeto que coincide con una de sus firmas de color. Así es como se ve PixyMon cuando Pixy está ejecutando el programa color\_connected\_components y ha detectado objetos:



*Ilustración 31. Programa color\_connected\_components detectando objetos (Imagen de Google)*

## 8. API de Arduino

### 8.1. Incluir archivos

Asegurarse de incluir estos dos archivos:

```
#incluir <SPI.h>
#include <Pixy2.h>
```

Se debe declarar una instancia de un objeto Pixy fuera de sus funciones **setup ()** y **loop ()**:

```
pixy2 pixie;
```

El método más importante en la biblioteca de Arduino es **getBlocks ()**, que devuelve la cantidad de objetos que ha detectado Pixy. Luego puede buscar en la **pixy.ccc.blocks []** matriz información sobre cada objeto detectado (un miembro de la matriz para cada objeto detectado). Cada miembro de la matriz (i) contiene los siguientes campos:

#### **uint16\_t getBlocks(uint16\_t maxBlocks=1000)**

- **pixy.ccc.blocks[i].m\_signature** El número de firma del objeto detectado (1-7 para firmas normales)
- **pixy.ccc.blocks[i].m\_x** La ubicación x del centro del objeto detectado (0 a 316)
- **pixy.ccc.blocks[i].m\_y** La ubicación y del centro del objeto detectado (0 a 208)
- **pixy.ccc.blocks[i].m\_width** El ancho del objeto detectado (1 a 316)
- **pixy.ccc.blocks[i].m\_height** La altura del objeto detectado (1 a 208)
- **pixy.ccc.blocks[i].m\_angle** El ángulo del objeto detectado objeto si el objeto detectado es un código de color (-180 a 180).
- **pixy.ccc.blocks[i].m\_index** El índice de seguimiento del bloque.
- **pixy.ccc.blocks[i].m\_age** El número de fotogramas que se ha rastreado el bloque.
- **pixy.ccc.blocks[i].print()** Una función miembro que imprime la información del objeto detectado en el puerto serie

#### **int8\_t setServos(uint16\_t s0, uint16\_t s1)**

Este método establece los servos de giro/inclinación que están conectados a los dos puertos de servo de Pixy. Los dos argumentos **s0** y **s1** pueden variar de 0 a 1000.

#### **int8\_t setBrillo (brillo uint8\_t)**

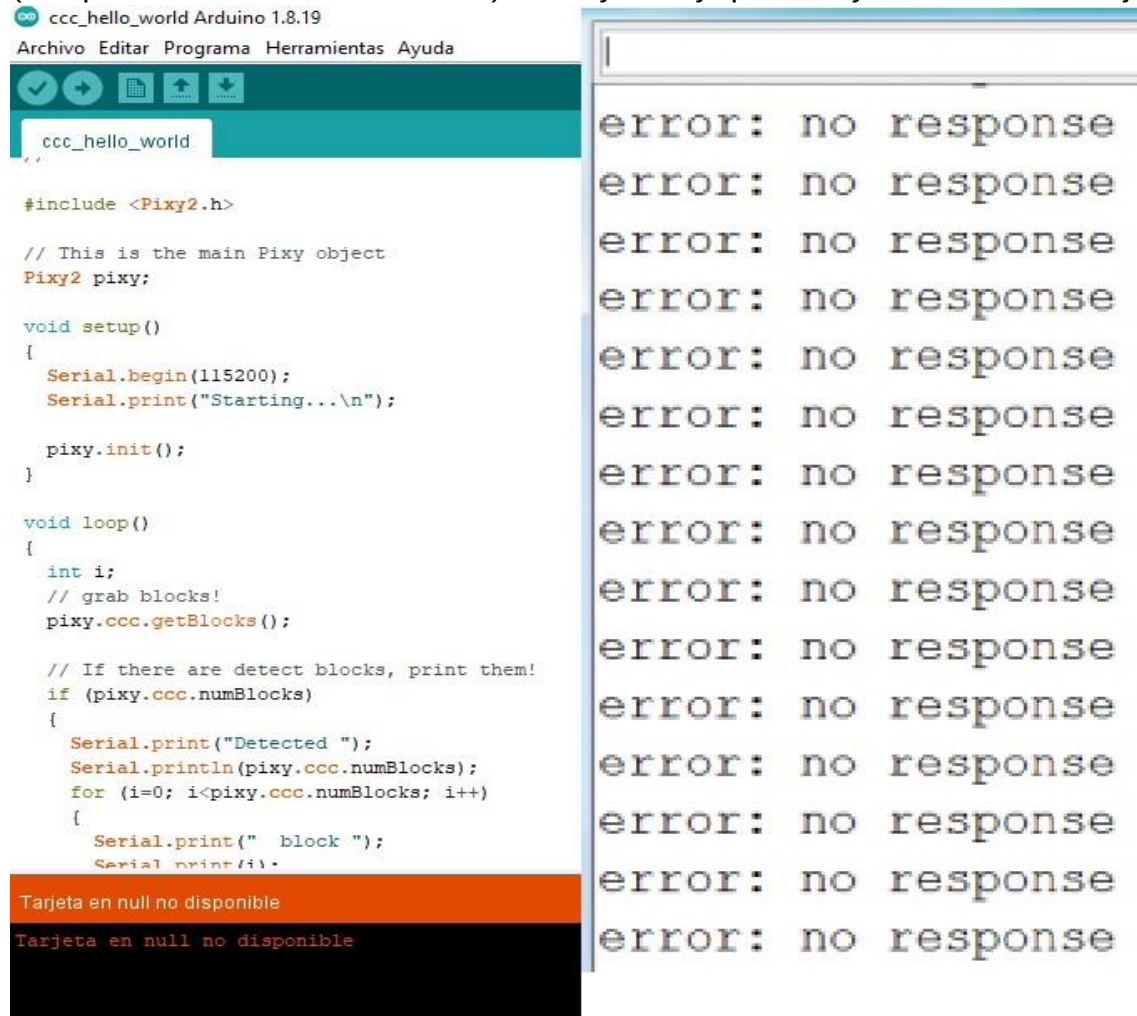
Este método establece el brillo (exposición) de la cámara de Pixy. El **brightness** argumento puede oscilar entre 0 y 255, siendo 255 la configuración más brillante.

#### **int8\_t setLED(uint8\_t r, uint8\_t g, uint8\_t b)**

Este método establece el LED RGB en la parte frontal de Pixy. Los argumentos **r** y **b** pueden oscilar entre 0 y 255.

## 8.2. Solución de problemas

Si se recibe el mensaje "error: no hay respuesta" del monitor serie Arduino, primero nos aseguraremos de que nuestra cámara esté ejecutando el programa ccc (componentes conectados a color) de PixyMon, y que le haya enseñado un objeto.



The screenshot shows the Arduino IDE interface. The top bar indicates 'ccc\_hello\_world Arduino 1.8.19'. The menu bar includes 'Archivo', 'Editar', 'Programa', 'Herramientas', and 'Ayuda'. The toolbar contains icons for checking, uploading, and downloading. The file name 'ccc\_hello\_world' is shown in the editor. The code in the editor is as follows:

```
#include <Pixy2.h>

// This is the main Pixy object
Pixy2 pixy;

void setup()
{
  Serial.begin(115200);
  Serial.print("Starting...\n");

  pixy.init();
}

void loop()
{
  int i;
  // grab blocks!
  pixy.ccc.getBlocks();

  // If there are detect blocks, print them!
  if (pixy.ccc.numBlocks)
  {
    Serial.print("Detected ");
    Serial.println(pixy.ccc.numBlocks);
    for (i=0; i<pixy.ccc.numBlocks; i++)
    {
      Serial.print("  block ");
      Serial.print(i);
    }
  }
}
```

At the bottom of the IDE, there are two error messages: 'Tarjeta en null no disponible' (Card is null not available) and 'Tarjeta en null no disponible' (Card is null not available). The serial monitor on the right shows the output: 'error: no response' repeated 15 times.

Ilustración 32. Error en Arduino en Hello world

## 9. L298N Controlador de Motores

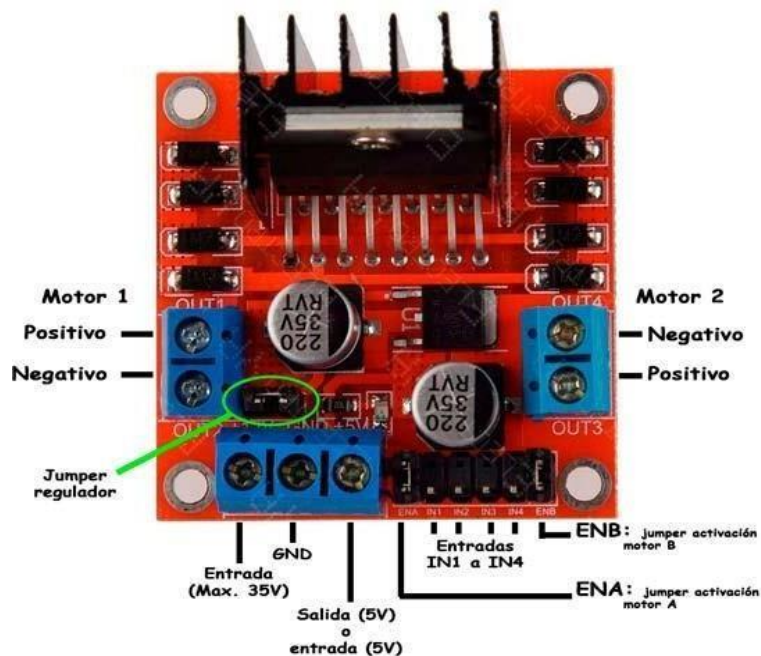


Ilustración 33. L298N Estructura de controlador de motores (Imagen de Google)

### 9.1. ¿Qué es L298N?

Este módulo posee dos puentes H que permiten controlar 2 motores DC o un motor paso a paso bipolar/unipolar.

El módulo permite controlar el sentido de giro y velocidad mediante señales TTL que se pueden obtener de microcontroladores y tarjetas de desarrollo como Arduino, Raspberry Pi o Launchpads de Texas Instruments.

### 9.2. Conexión y funcionamiento

Incorpora una entrada de voltaje. El rango de tensiones en el que trabaja este módulo va desde 3V hasta 35V, dependiendo de la configuración del Jumper regulador; y cuenta con una intensidad de hasta 2A. Con el Jumper puesto sirve para controlar motores de 5 a 12V y tendremos una salida de 5V para otros módulos. Sin el Jumper regulador, sirve para controlar motores de 12 a 35V y en este caso, el tercer pin actuará como una entrada de 5V para alimentar la parte lógica del driver.

Las salidas para los motores 1 y 2 nos darán la energía para mover los motores. Tener en cuenta la polaridad al conectarlos, para que cuando más tarde hagamos que se muevan, funcionen como deberían. Si no tendríamos solo que invertir las conexiones.

Los pines **IN1 e IN2** nos sirven para controlar el sentido de giro del motor 1, y los pines **IN3 e IN4** el del motor 2. Funcionan de forma que, si IN1 está a HIGH e IN2 a LOW, el motor 1 gira en un sentido, y si está IN1 a LOW e IN2 a HIGH lo hace en el otro. Y lo mismo con los pines IN3 e IN4 y el motor 2.

- IN1 -> HIGH; IN2 -> LOW: gira en un sentido
- IN1 -> LOW; IN2 -> HIGH: gira en el sentido contrario

**Para controlar la velocidad de giro de los motores tenemos que quitar los jumpers y usar los pines ENA y ENB.** Los conectaremos a dos salidas PWM de Arduino de forma que le enviemos un valor entre 0 y 255 que controle la velocidad de giro. Si tenemos los jumpers colocados, los motores girarán siempre a la misma velocidad.

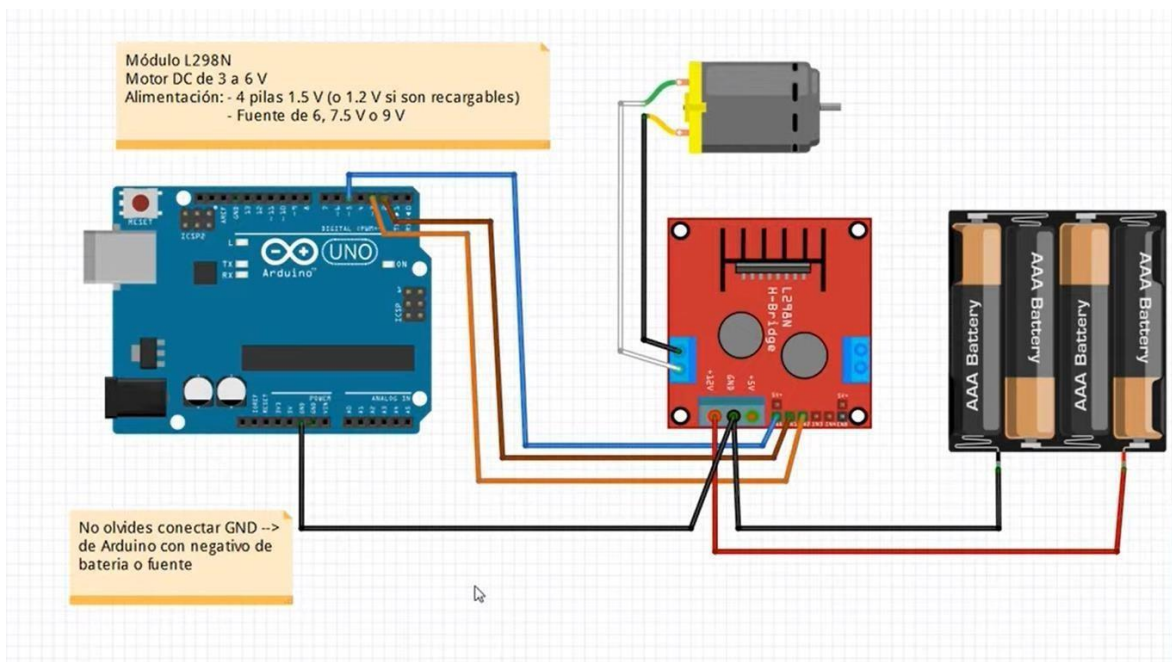


Ilustración 34. Conexión de Arduino con L298N (Imagen de Google)

### 9.3. Motor de corriente continua

Es una máquina que convierte energía eléctrica en energía mecánica, provocando un movimiento rotatorio, gracias a la acción de un campo magnético.

Los componentes de un motor de corriente continua se dividen en dos partes:

- **Estator:** parte que da soporte mecánico al aparato y contiene los polos de la máquina, que pueden ser devanados de hilo de cobre sobre un núcleo de hierro o imanes permanentes.

- **Rotor:** es un componente generalmente de forma cilíndrica, también devanado y con núcleo, alimentado con corriente continua a través del colector formado por delgas. Las delgas se fabrican generalmente de cobre y están en contacto alternante con las escobillas fijas.

Un motor de corriente continua permite regular continuamente la velocidad del eje y obtener un par de arranque elevado. Para ello, es necesario aplicar una corriente continua tanto en el inducido como en el inductor.

- Al conectar el motor de la fuente de alimentación la corriente circula por las bobinas inductoras creando un electroimán y creando el campo magnético que necesitamos.
- Esta corriente continua también circula por las bobinas inducidas a través las escobillas y el colector.
- Una vez establecido el campo magnético los pares de fuerza que actúan sobre las bobinas inducidas la obligaran a girar y con ellas girará todo el motor.

## 10. Resultados

### 10.1. Construcción del robot móvil con Arduino y Cámara Pixy

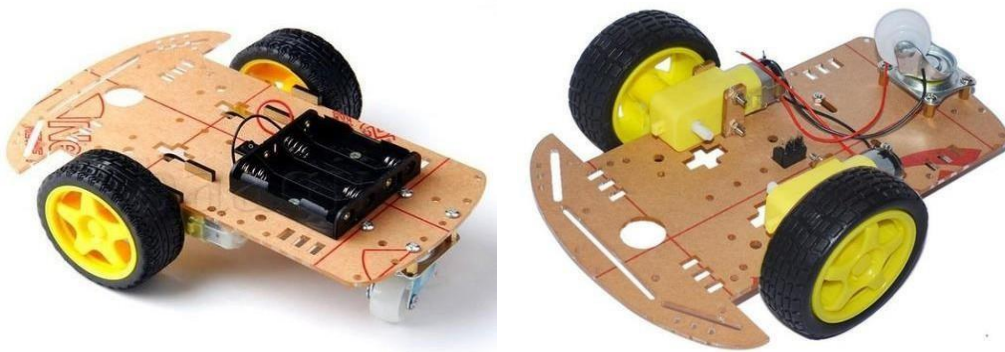


Ilustración 35. Chasis Auto Robot

### 10.2. Cableado del motor L298N a Arduino

Conectaremos la fuente de alimentación externa de 12V al terminal VS. Teniendo en cuenta la caída de voltaje interno del L298N IC, los motores recibirán 10 V y girarán a RPM ligeramente más bajas.

Luego, necesitamos suministrar 5V al circuito lógico del L298N. Usaremos el regulador de 5V incorporado para obtener 5V de la fuente de alimentación del motor.

Conectamos los pines de entrada y habilitación del módulo L298N (ENA, IN1, IN2, IN3, IN4 y ENB) a los seis pines de salida digital Arduino.

| PINES DE ENTRADA | PINES DE SALIDA |
|------------------|-----------------|
| ENA              | 5               |
| IN1              | 2               |
| IN2              | 3               |
| IN3              | 7               |
| IN4              | 8               |
| ENB              | 9               |

Finalmente conectamos un motor al terminal **A (OUT1 y OUT2)** y el otro motor al terminal **B (OUT3 y OUT4)**.

En esta imagen se muestran las conexiones.

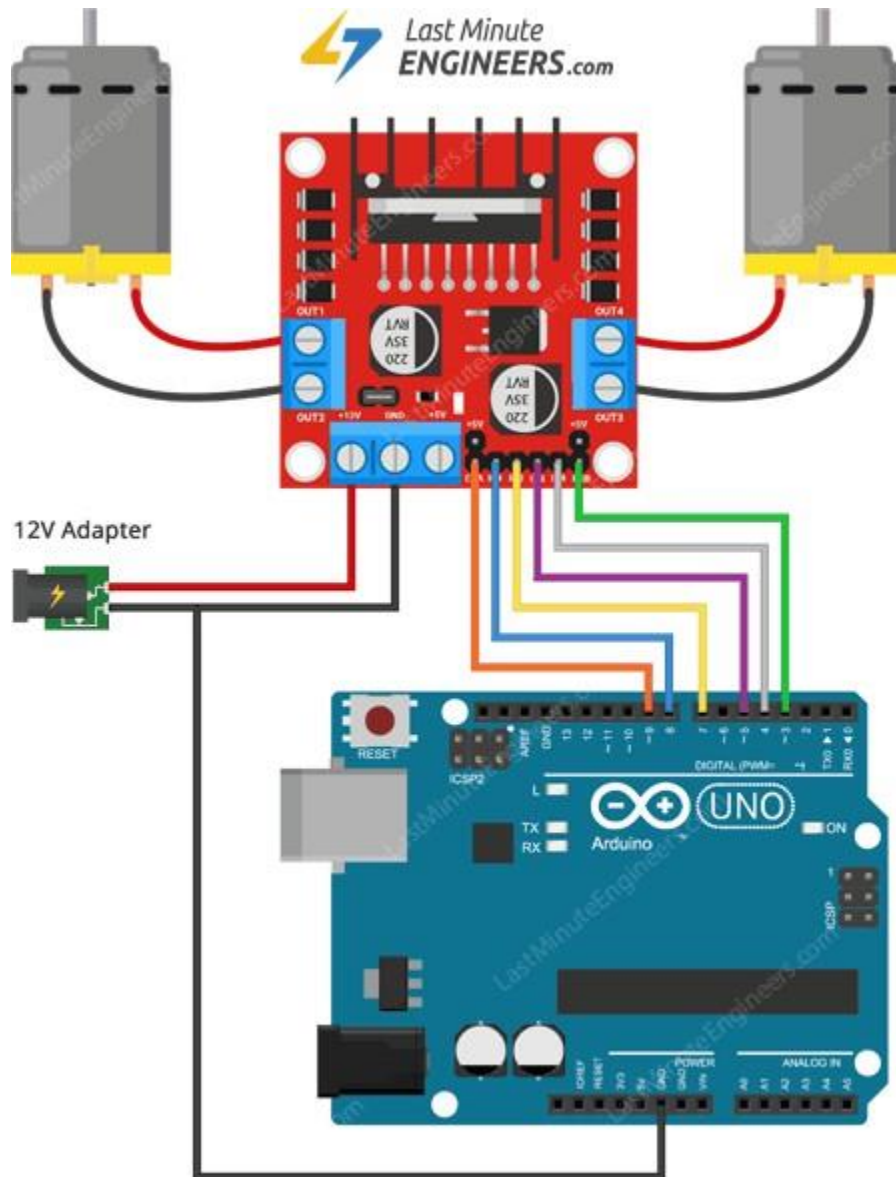
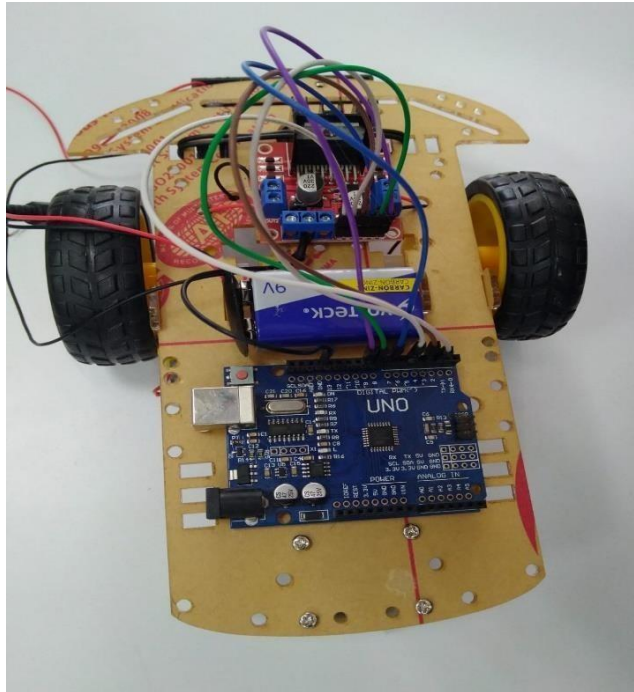
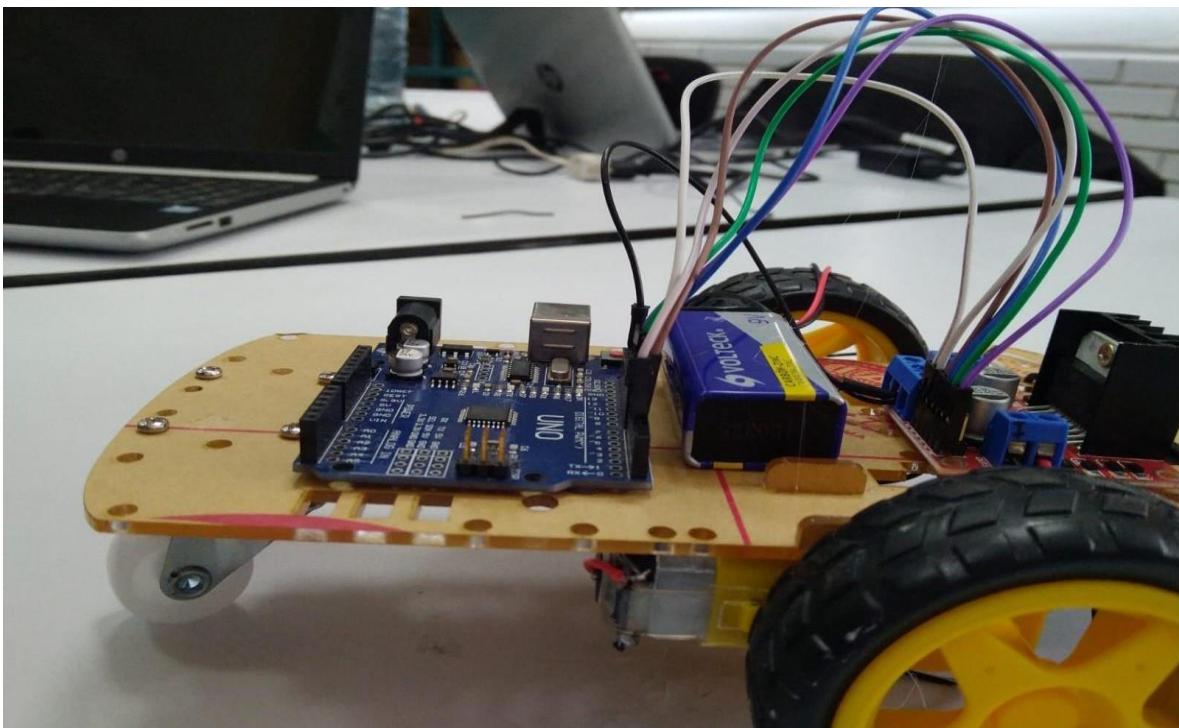


Ilustración 36. Esquema de conexión L298N a Arduino

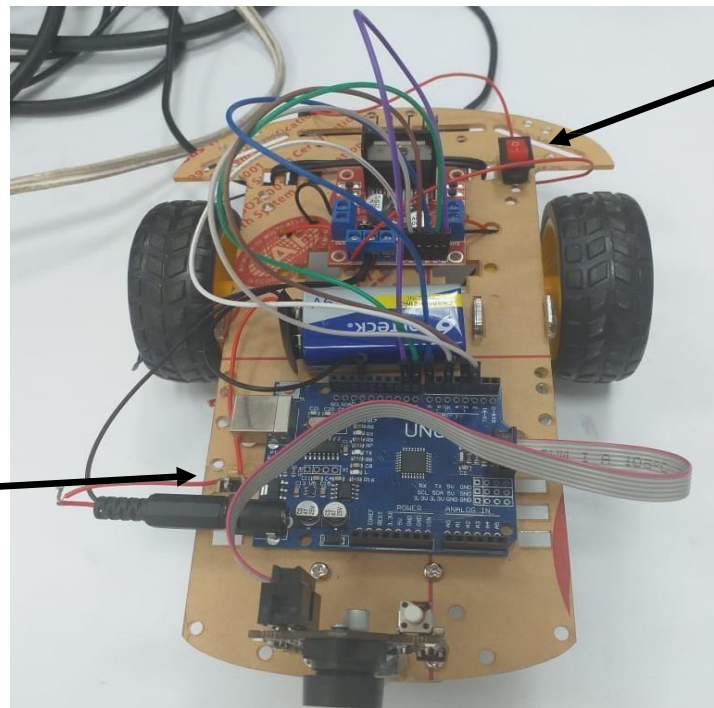
Estas son fotografías del prototipo ya armado con la tarjeta L298N y Arduino y las respectivas conexiones con los pines de entrada y salida y la conexión de los motores CC.



*Ilustración 37. Prototipo de Robot Móvil, vista por arriba*

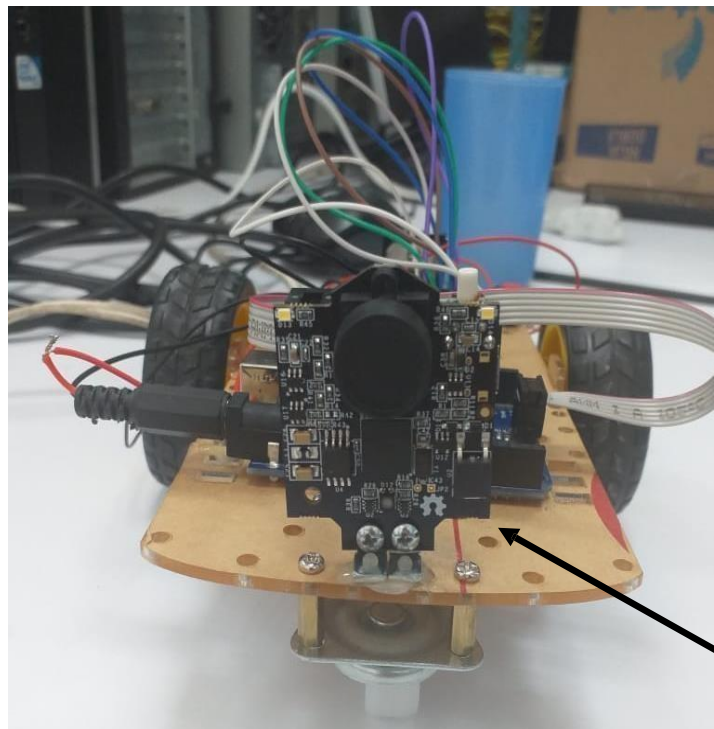


*Ilustración 38. Prototipo armado con Arduino y L298N, vista de lado*



Switch para controlar el apagado y encendido de la pila para el Arduino.

Switch para controlar el apagado y encendido de las pilas para los motores



Cámara Pixy montada en el chasis Auto Robot para detectar los objetos.

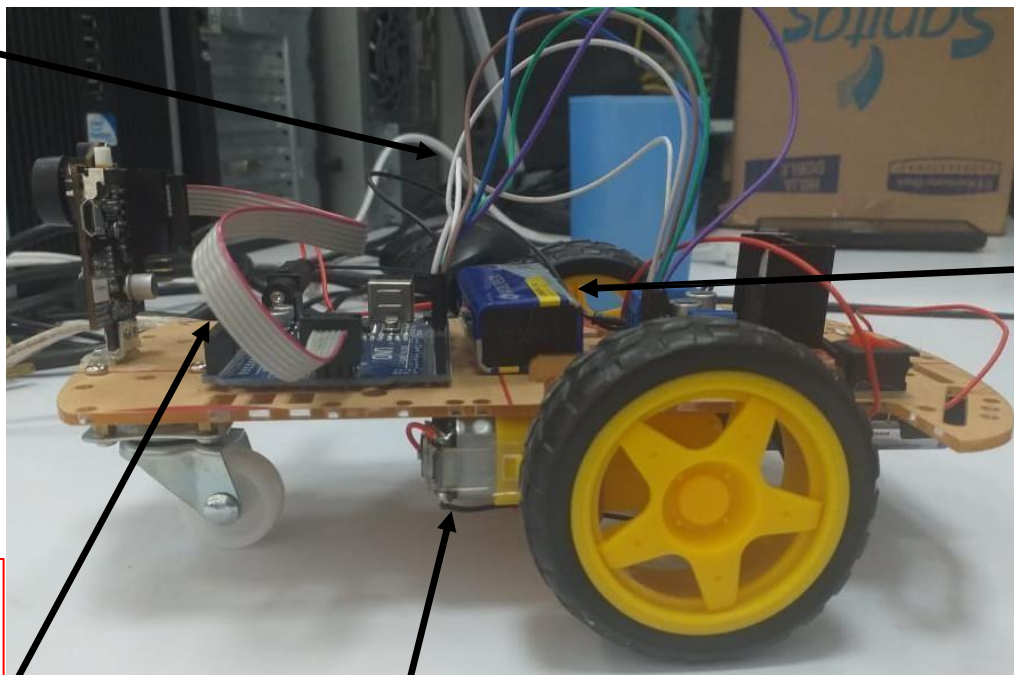
*Ilustración 39. Imágenes del Robot para detectar objetos*

Pines de entrada conectados con la tarjeta L298N

Tarjeta Arduino UNO conectada a la Cámara Pixy con el cable FC-10P

Pila de 9V para Arduino.

Motores del Chasis Auto Robot, para poder ir a donde esta el objeto identificado por la Cámara Pixy.



*Ilustración 40. Robot para detectar Objetos*

### 10.3. Diagrama de flujo

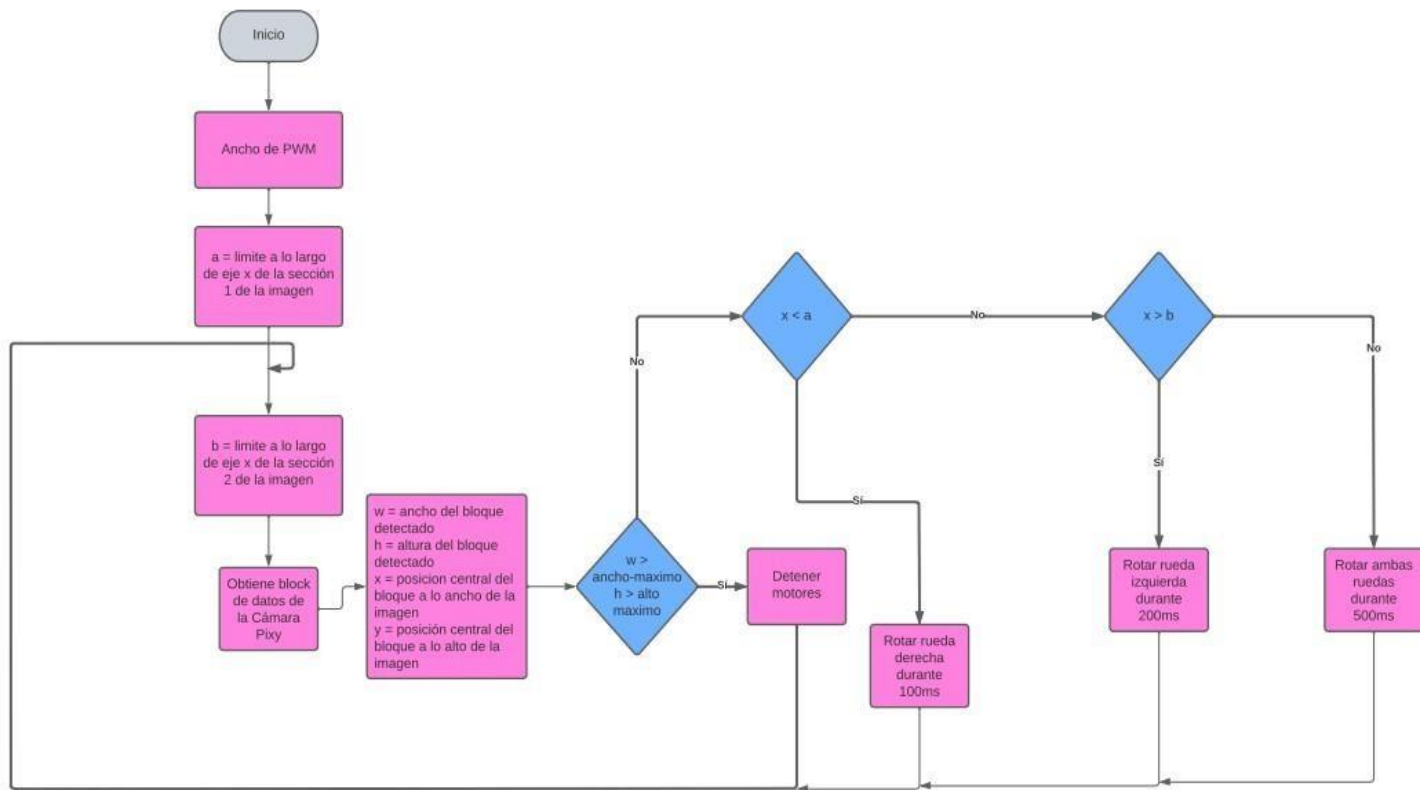


Ilustración 41. Diagrama de flujo del Robot Móvil

## 10.4. Código en Arduino

Comienza con la declaración de los pines de Arduino a los que se conectan los pines de control del L298N.

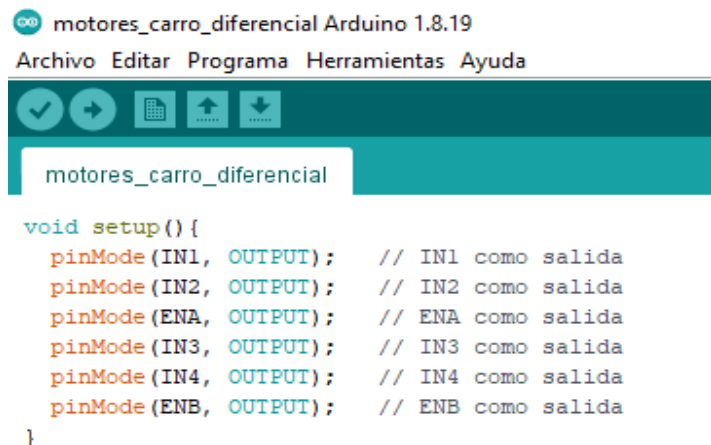


The screenshot shows the Arduino IDE interface with the file name 'motores\_carro\_diferencial' and version '1.8.19'. The menu bar includes 'Archivo', 'Editar', 'Programa', 'Herramientas', and 'Ayuda'. The toolbar contains icons for checking, running, serial monitor, and file operations. The code editor displays the following declarations:

```
int IN1 = 2;      // IN1 a pin digital 2
int IN2 = 3;      // IN2 a pin digital 3
int ENA = 5;      // ENA a pin digital 5
int IN3 = 7;      // IN3 a pin digital 7
int IN4 = 8;      // IN4 a pin digital 8
int ENB = 9;      // ENA a pin digital 9
int VELOCIDAD;    // variable para almacenar valor de velocidad
```

Ilustración 42. Código en Arduino parte 1

En la sección de configuración del código, todos los pines de control del motor (tanto los pines de control de dirección como de velocidad) se declaran como SALIDA digital.



The screenshot shows the same Arduino IDE interface as the previous one, but with the 'void setup()' function defined. The code editor displays the following configuration:

```
void setup() {
  pinMode(IN1, OUTPUT); // IN1 como salida
  pinMode(IN2, OUTPUT); // IN2 como salida
  pinMode(ENA, OUTPUT); // ENA como salida
  pinMode(IN3, OUTPUT); // IN3 como salida
  pinMode(IN4, OUTPUT); // IN4 como salida
  pinMode(ENB, OUTPUT); // ENB como salida
}
```

Ilustración 43. Código en Arduino parte 2

En la sección de bucle del código llamamos a tres funciones definidas por el usuario.



```
void loop() {  
    VELOCIDAD = 200;    // valor de velocidad para PWM de ENA y ENB  
  
    aAvance(VELOCIDAD); // funcion de avance del motor A  
    bAvance(VELOCIDAD); // funcion de avance del motor B  
    delay(3000);        // demora de 3 seg.  
  
    detieneAmbos();     // funcion que detiene ambos motores  
    delay(2000);        // demora de 2 seg.  
  
    aAvance(VELOCIDAD); // funcion de avance del motor A (giro a derecha)  
    delay(1000);        // demora de 1 seg.  
  
    detieneAmbos();     // funcion que detiene ambos motores  
    delay(2000);        // demora de 2 seg.  
  
    aAvance(VELOCIDAD); // funcion de avance del motor A  
    bAvance(VELOCIDAD); // funcion de avance del motor B  
    delay(3000);        // demora de 3 seg.  
  
    detieneAmbos();     // funcion que detiene ambos motores  
    delay(2000);        // demora de 2 seg.  
  
    aAvance(VELOCIDAD); // funcion de avance del motor A (giro a derecha)  
    delay(1000);        // demora de 1 seg.  
  
    detieneAmbos();     // funcion que detiene ambos motores  
    delay(2000);        // demora de 2 seg.  
}  
  
detieneAmbos(); // funcion que detiene ambos motores  
delay(2000);    // demora de 2 seg.  
  
aAvance(VELOCIDAD); // funcion de avance del motor A (giro a derecha)  
delay(1000);        // demora de 1 seg.  
  
detieneAmbos(); // funcion que detiene ambos motores  
delay(2000);        // demora de 2 seg.  
  
aAvance(VELOCIDAD); // funcion de avance del motor A (giro a derecha)  
delay(1000);        // demora de 1 seg.  
  
detieneAmbos(); // funcion que detiene ambos motores  
delay(2000);        // demora de 2 seg.  
  
aAvance(VELOCIDAD); // funcion de avance del motor A  
bAvance(VELOCIDAD); // funcion de avance del motor B  
delay(3000);        // demora de 3 seg.  
  
detieneAmbos(); // funcion que detiene ambos motores  
delay(2000);        // demora de 2 seg.  
}
```

Ilustración 44. Código en Arduino parte 3

Estas funciones son :

**aAvance(VELOCIDAD)** : Esta funcion hace que el motor A avance durante (¿?segundos).

**bAvance(VELOCIDAD)** : Esta funcion hace que el motor B avance durante (¿? Segundos).

**detieneAmbos** : Esta funcion hace que ambos motores se detengan durante (¿?segundos).

#### 10.4.1. Explicación de lo que hace el código

El motor A y B avanzan durante 3 segundos, se detienen 2 segundos, el motor A avanza 1 segundo y gira hacia la derecha, se detiene 2 segundos despues el motor A y B vuelve a avanzar 3 segundo y a detenerse por 2 segundos; el motor vuelve a avanzar 1 segundo y gira a la derecha, se detiene 2 segundos, luego avanza 3 segundos y se detiene 2 segundos , esto para formar un cuadrado.

```
void aAvance(int veloc){ // funcion para avance de motor A
  analogWrite(ENA, veloc); // velocidad mediante PWM en ENA
  digitalWrite(IN1, LOW); // IN1 a cero logico
  digitalWrite(IN2, HIGH); // IN2 a uno logico
}

void bAvance(int veloc){ // funcion para avance de motor B
  analogWrite(ENB, veloc); // velocidad mediante PWM en ENB
  digitalWrite(IN3, LOW); // IN3 a cero logico
  digitalWrite(IN4, HIGH); // IN4 a uno logico
}

void detieneAmbos(){ // funcion que detiene ambos motores
  analogWrite(ENA, 0); // deshabilita motor A
  analogWrite(ENB, 0); // deshabilita motor B
}
```

---

*Ilustración 45. Código en Arduino parte 4*

## 11. Pruebas

| No. De Prueba | Descripción                                                                                                                                                                                                                                                                            | Ventaja                                                                                                                                                                                                                                                                                             | Desventaja                                                                                                                                                  |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1             | Cuando se hizo la primera prueba del robot, todavía no tenía integrada la cámara Pixy, solo fue una prueba para ver el funcionamiento de las llantas.                                                                                                                                  | Se tuvo la ventaja de poder observar como el programa en Arduino hacía que las llantas giraran y formaran un cuadrado y observar si había alguna falla en el programa.                                                                                                                              | La desventaja fue que una de las llantas estaba muy pegada al chasis, y eso hacía que no avanzara bien.                                                     |
| 2             | Se hizo la segunda prueba e igualmente sin la cámara y en esta ocasión no formo el cuadrado, sino que avanzaba derecho y después daba vuelta y volvía a dar vuelta sin formar el cuadrado.                                                                                             | Siento que una ventaja fue que con esa prueba pudimos saber el funcionamiento de las llantas, los motores, la tarjeta Arduino y L298N.                                                                                                                                                              | La desventaja fue que una de las llantas estaba muy pegada al chasis, y eso hacía que no avanzara bien.                                                     |
| 3             | En esta prueba ya estaba implementada la Cámara Pixy, le pusimos un Objeto a la cámara para que lo reconociera, cuando lo reconoció le cargamos el programa a la tarjeta Arduino y lo probamos; en este caso le poníamos el objeto a una distancia y este lo reconocía e iba hacia él. | Que pudo encontrar el objeto y buscarlo por su propia cuenta.                                                                                                                                                                                                                                       | Tenía como desventaja que cuando reconocía el objeto y llegaba a él; no se detenía, se seguía derecho llevándose con él.                                    |
| 4             | En esta prueba le ajustamos para que pudiera detenerse al llegar al objeto que detecto.                                                                                                                                                                                                | Cuando se está detectando el objeto por la Cámara Pixy, estamos usando una tabla que se llama Matriz Blocks y en esa tabla se refleja la distancia en la que se encuentra el objeto y con esa distancia es la que usamos en el programa Arduino, poniendo una distancia en específico de un objeto. | Que como fue una distancia específica, se alejaba el objeto y no alcanzaba a detectarlo.                                                                    |
| 5             | Aquí en esta prueba, hice tres figuras geométricas; un cuadrado, un triángulo y un rectángulo y las pusimos frente a la Cámara para que las reconociera.                                                                                                                               | La cámara si las reconoció y el robot móvil si las detectaba se detenía al encontrarlas.                                                                                                                                                                                                            | Un problema que se tuvo es que como las figuras eran muy pequeñas, quedaban debajo de la cámara, teníamos que alzarlas con la mano para que pudiera verlas. |

|   |                                                                                      |                                                                                                                                                          |                                                                                                                                                                                                                              |
|---|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 6 | En esta última prueba, probamos con un objeto más grande, en este caso un vaso azul. | Cuando lo pusimos a que detectara el objeto, ya lo hacía mucho mejor porque estaba un poco más alto y entraba por así decirlo en el sensor de la cámara. | La desventaja que tuvo y que fue en todas las pruebas, era la luz, el objeto tenía que estar en un lugar con mucha luz para que Pixy viera mejor el objeto, entonces, entre menos luz hubiera, menos se detectaba el objeto. |
|---|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

### 11.1. Imágenes de las pruebas



Ilustración 46. Mostrándole el Objeto al Robot Móvil

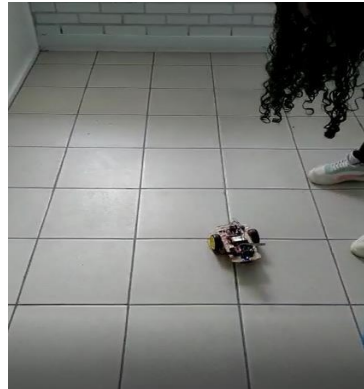


Ilustración 47. Robot Móvil buscando el Objeto



Ilustración 50. Siguiendo el Objeto



Ilustración 51. Siguiendo el Objeto hacia donde lo mueva

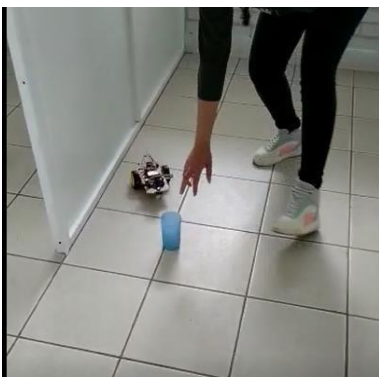


Ilustración 48. Deteniéndose cuando llega al objeto



Ilustración 49. Llegando al objeto

## 12. Conclusiones

### 12.1. Conclusiones de proyecto

En la realización de este proyecto que tuvo como objetivo desde un principio o al menos una idea era crear un pequeño robot móvil para fines educativos, tal como el que sirviera para poder ayudar a los niños a reconocer las figuras geométricas de una manera divertida para ellos; también así poder despertar en ellos un interés hacia la tecnología ya que hoy en día es una parte fundamental en la vida de cada persona ya que la tecnología cada día va creciendo y se van creando cosas que nos facilitan más la vida.

La Cámara Pixy es muy capaz, no solo se puede crear un robot móvil para reconocer objetos, sino que su capacidad puede lograr muchas más cosas en un futuro para ayudar a mejorar cosas tan simples pero que en realidad hacen mucho; como, detectar códigos de barra, intersecciones y seguimientos de línea, ya que podemos posicionar colores u objetos con sólo marcarlos a través del software PixyMon.

También se comprobó que la cámara es muy fácil de manejar así como su instalación y el fácil reconocimiento de objetos con solo ponerlos enfrente de la cámara y al apretar un botón para que queden grabados los objetos, así como también la interactividad que tiene con la placa de Arduino.

### 12.2. Recomendaciones

El robot móvil creado con la cámara Pixy y con la tarjeta Arduino se puede mejorar y hacerle cambios ya que solo no sirva para fines educativos, sino que vaya más allá de sus capacidades, ya que esta cámara cuenta con sensor de movimiento que con tener eso es capaz de hacer otra cosa y no solo reconocer objetos geométricos para la enseñanza educativa.

### 12.3. Experiencia personal y profesional adquirida

Al escoger trabajar en este proyecto, en lo personal, fue en primera porque siempre me ha llamado la atención la robótica, así que lo escogí para empezar a adentrarme un poco a ella con este pequeño robot móvil que me pareció muy interesante en cómo se iba a realizar que en este caso fue con una cámara Pixy que de igual forma no conocía que existía y que funcionalidades tenía.

También al estar en este proyecto aprendí a informarme, a buscar información de diferentes fuentes y que el tiempo es muy importante para la realización de un proyecto.

En la experiencia profesional adquirida, aprendí que para la realización de un proyecto es muy importante estar enfocados en lo que vamos a hacer, el tiempo con el que se tiene planteado un proyecto a veces se puede tomar más tiempo del previsto si no se hace una planificación o un cronograma sobre las actividades a realizar y lo importante de hacerlas es seguirlas.

También si no entiendo algo o no me sale algo, es muy importante tener comunicación con el equipo de trabajo para lograr un desarrollo constante y ordenado.

### **13. Competencias desarrolladas**

- Conocer y aplicar conocimientos de ciencias y tecnologías básicas a la práctica de la Robótica.
- Poseer capacidad para diseñar, desarrollar, implementar, gestionar y mejorar productos, sistemas y procesos en los distintos ámbitos relacionados con la robótica usando técnicas analíticas, computacionales o experimentales.
- Aplicar los conocimientos adquiridos para identificar, formular y resolver problemas dentro de contextos amplios y multidisciplinarios, siendo capaces de integrar conocimientos, trabajando en equipos multidisciplinarios.
- Comprender el impacto de la Robótica en el medio ambiente, el desarrollo sostenible de la sociedad y la importancia de trabajar en un entorno profesional y responsable.
- Poseer habilidades de aprendizaje que permitan continuar estudiando a lo largo de la vida para su adecuado desarrollo profesional.
- Incorporar nuevas tecnologías y herramientas en la Robótica en sus actividades profesionales.
- Capacidad para generar nuevas ideas (creatividad).

#### 14. Fuentes de información

- pixycam. (2018). Color Connected Components. (Marzo 15, 2022, de PIXY Sitio web: [https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:color\\_connected\\_components](https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:color_connected_components))
- pixycam. (2020). Conexión de Pixy a un microcontrolador (como un Arduino). (Marzo 25, 2022), de PIXY Sitio web: [https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:hooking\\_up\\_pixy\\_to](https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:hooking_up_pixy_to_a_microcontroller_-28like_an_arduino-29)

#### a microcontroller -28like an arduino-29

- Library. (2018). PixyMon. (Abril 10, 2022), de Library Sitio web: <https://1library.co/article/pixymon-dise%C3%B1o-implementaci%C3%B3n-robot-personal-transportador-objetos.yr36jdoy>
- (Guerra, C.L., Morochó, B.O., & Marlon, A.). (2016). **DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ROBÓTICO MÓVIL COOPERATIVO PARA DETECCIÓN Y ANÁLISIS DE INCENDIOS EN AMBIENTES CONTROLADOS.** (Mayo 25, 2022), de ESPE Sitio web: <http://repositorio.espe.edu.ec/jspui/bitstream/21000/12465/1/T-ESPE-053580.pdf>
- Fernandez, Y.. (2020). Qué es Arduino, cómo funciona y qué puedes hacer con uno. (Junio 18, 2022), de Xataka Basics Sitio web: <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>
- pixycam. (2021). Descargas. (Febrero 24, 2022), de PIXY Sitio web: <https://pixycam.com/downloads-pixy2/>
- Wordpress. (2021). ICSP. Julio 8, 2022, de Wordpress Sitio web: <https://aprendiendoarduino.wordpress.com/2016/11/06/icsp/>
- HETPRO. (2021). L293D – Controlador de motores. (Agosto 01, 2022), de HETPRO Sitio web: <https://hetpro-store.com/TUTORIALES/l293d/>
- DroneBot Workshop. (2019). Pixy2 Camera - Image Recognition for Arduino & Raspberry Pi. Marzo 15, 2022, de DroneBot Workshop Sitio web: <https://www.youtube.com/watch?v=391dXDjqzXA&t=2038s>
- Microcontrollres Projects. (2017). Pixy Cam to with Arduino. Junio 20, 2022, de Microcontrollres Projects Sitio web: [https://www.youtube.com/watch?v=DV4YK\\_Kk5IY&t=50s](https://www.youtube.com/watch?v=DV4YK_Kk5IY&t=50s)



Instituto Tecnológico de Apizaco  
Sistemas y Computación

Apizaco, Tlaxcala, **14/octubre/2022**

OFICIO No. S.C./859/2022

ASUNTO: Carta de liberación

**NORMA ANGÉLICA AGUIÑAGA SÁNCHEZ**  
**JEFA DEL DEPTO. DE GESTIÓN TECNOLÓGICA Y VINCULACIÓN**  
**PRESENTE**

Por medio de la presente, me dirijo a usted para comunicar que el (la) C. **ZAIRA HAZEL CRUZ BREMONT** con número de control **16371136** estudiante de la Carrera de **INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES**, ha concluido satisfactoriamente su Residencia Profesional, desarrollando el proyecto **"PROGRAMA DE SEGUIMIENTO DE OBJETOS UTILIZANDO UNA CÁMARA PIXYCAM"**. En el Instituto Tecnológico de Apizaco.

Sin más por el momento hago propicia la ocasión para enviarle un cordial saludo.

**ATENTAMENTE**

*Excelencia en Educación Tecnológica®  
Pensar para Servir, Servir para Triunfar®*

  
**SAÚL OLAF LOATZA MELÉNDEZ**  
**JEFE DEL DEPTO. DE SISTEMAS Y COMPUTACIÓN**

  
SECRETARÍA DE EDUCACIÓN PÚBLICA  
TECNOLÓGICO NACIONAL  
DE MÉXICO  
INSTITUTO TECNOLÓGICO DE APIZACO  
DEPARTAMENTO DE  
SISTEMAS Y COMPUTACIÓN

ccp. Archivo

SOLM/Chely\*



Carretera Apizaco-Tzompantepec, esquina con Av. Instituto Tecnológico S/N,  
Conurbado Apizaco-Tzompantepec, Tlaxcala, Mex. C.P. 90491 Tel. (241) 4172010 Ext. 115,  
e-mail: sistemas@apizaco.tecnm.mx | tecnm.mx | apizaco.tecnm.mx



**2022 Flores**  
Año de  
**Magón**  
PRECURSOR DE LA REVOLUCIÓN MEXICANA



Apizaco, Tlaxcala, **14/octubre/2022**

OFICIO No. S.C./860/2022

ASUNTO: Carta de No Inconveniencia

**SERGIO OCTAVIO ROSALES AGUAYO**  
**DIRECTOR**  
**PRESENTE**

Por medio de la presente me permito dirigirme a usted, para informarle que no existe inconveniente alguno para que la C. **Zaira Hazel Cruz Bremont** con número de control **16371136** estudiante de la carrera de **Ingeniería en Tecnologías de la Información y Comunicaciones**, presente y haga uso de la información que proporcione en su proceso de residencias profesionales, contenida en el proyecto denominado "**Programa de seguimiento de objetos utilizando una cámara pixycam**", en el **Instituto Tecnológico de Apizaco**, durante periodo establecido, por lo que, no hay ningún inconveniente para que dicho proyecto sea utilizado única y exclusivamente en el proceso de titulación por la opción **Titulación Integral Informe de Residencia Profesional**, confiando en que la presentación de la información del proyecto se realizará de manera responsable.

Se expide la presente carta de no inconveniencia para los trámites a que haya lugar y sin otro particular por el momento, aprovecho la ocasión para enviarle un cordial saludo.

**ATENTAMENTE**

*Excelencia en Educación Tecnológica®*  
*Pensar para Servir, Servir para Triunfar®*

  
**SAÚL OLAF LOAIZA MELÉNDEZ**  
**JEFE DEL DEPTO. DE SISTEMAS Y COMPUTACIÓN**



ccp. Archivo

SOLM/Chely\*

