Programa de Infraestructura del Transporte (PITRA)

Informe No. LM-PI-USVT-015-19

Informe Final

Proyecto:

Análisis exploratorio de la detección de objetos mediante un Arduino Uno y una PixyCam

Preparado por:

Unidad de Seguridad Vial y Transporte

San José, Costa Rica
Diciembre, 2019

1. Informe LM-PI-USVT-015-19	**2. Copia No.** 1
3. Título y subtítulo: Análisis exploratorio de la detección de objetos mediante un Arduino Uno y una PixyCam.	**4. Fecha del Informe** 10 de diciembre, 2019

7. Organización y dirección

Laboratorio Nacional de Materiales y Modelos Estructurales
Universidad de Costa Rica, Ciudad Universitaria Rodrigo Facio,
San Pedro de Montes de Oca, Costa Rica.
Tel: (506) 2511-2500 / Fax: (506) 2511-4440

8. Notas complementarias Informe elaborado por Bayron Blanco Alfaro, asistente de la Unidad de Seguridad Vial y Transporte del PITRA-Lanamme, bajo la supervisión de Henry Hernández Vega

9. Resumen

Este informe detalla las labores llevadas a cabo en el análisis exploratorio de la detección de objetos mediante la conexión de una camara PixyCam con una placa Arduino Uno, acompañados de una programación en la interfaz Arduino. El objetivo del proyecto es explorar el uso de la cámara PixyCam para la detección de objetos con el fin de evaluar su uso en aplicaciones de monitoreo del tráfico. A partir de los resultados obtenidos, se llega a la conclusión únicamente es capaz de reconocer objetos de ciertos colores y con cierta intensidad o brillo. Debido a las limitaciones del equipo evaluado no se considera adecuado para la detección y monitoreo en aplicaciones de ingeniería de tránsito.

10. Palabras clave PITRA, USVT, Arduino, PixyCam, PanTilt	**11. Nivel de seguridad:** Ninguno	**12. Núm. de páginas** 26

13. Elaborado por:	**14. Revisado por:**	
Ing. Henry Hernández Vega Unidad de Seguridad Vial y Transporte, PITRA-LanammeUCR HENRY HERNANDEZ VEGA (FIRMA) Firmado digitalmente por HENRY HERNANDEZ VEGA (FIRMA) Nombre de reconocimiento (DN): serialNumber=CPF-01-1169-0134, sn=HERNANDEZ VEGA, givenName=HENRY, c=CR, o=PERSONA FISICA, ou=CIUDADANO, cn=HENRY HERNANDEZ VEGA (FIRMA) Fecha: 2019.12.10 11:03:16 -06'00'	**Ing. Diana Jiménez Romero, MSc, MBA** Coordinadora Unidad de Seguridad Vial y Transporte, PITRA-LanammeUCR DIANA MARIA JIMENEZ ROMERO (FIRMA) Firmado digitalmente por DIANA MARIA JIMENEZ ROMERO (FIRMA) Nombre de reconocimiento (DN): serialNumber=CPF-01-0837-0128, sn=JIMENEZ ROMERO, givenName=DIANA MARIA, c=CR, o=PERSONA FISICA, ou=CIUDADANO, cn=DIANA MARIA JIMENEZ ROMERO (FIRMA) Fecha: 2019.12.10 11:33:46 -06'00'	**Lic. Miguel Chacón, MSc, MBA** Asesor Legal Externo LanammeUCR MIGUEL CHACON ALVARADO (FIRMA) Firmado digitalmente por MIGUEL CHACON ALVARADO (FIRMA) Fecha: 2019.12.10 12:17:56 -06'00'
Fecha: 10/12/ 2019	**Fecha: 10/12/ 2019**	**Fecha: 10/12/ 2019**

15. Aprobado por:		
Ing. Ana Luisa Elizondo Salas, MSc Coordinadora Programa Infraestructura del Transporte, LanammeUCR ANA LUISA ELIZONDO SALAS (FIRMA) Firmado digitalmente por ANA LUISA ELIZONDO SALAS (FIRMA) Nombre de reconocimiento (DN): serialNumber=CPF-06-0206-0266, sn=ELIZONDO SALAS, givenName=ANA LUISA, c=CR, o=PERSONA FISICA, ou=CIUDADANO, cn=ANA LUISA ELIZONDO SALAS (FIRMA) Fecha: 2019.12.10 14:18:19 -06'00'	**Ing. Alejandro Navas Carro, MSc.** Director LanammeUCR Firmado digitalmente por ALEJANDRO NAVAS CARRO (FIRMA) Fecha: 2019.12.11 15:43:54 -06'00'	
Fecha: 10/12/ 2019	**Fecha: 10/12/2019**	

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

Tabla de Contenidos

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

1. RESUMEN

Este informe detalla las labores llevadas a cabo en el análisis exploratorio de la detección de objetos mediante la conexión de una cámara PixyCam con una placa Arduino Uno, acompañados de una programación en la interfaz Arduino. Se expone sobre los productos que ofrece Arduino, su interfaz de programación y su monitor serie. También, se detallan los productos ofrecidos por Pixy, el funcionamiento de la PixyCam con el programa PixyMon y una base para la cámara llamado Pan/Tilt.

Luego se muestra la conexión entre la placa Arduino Uno y la cámara PixyCam y su funcionamiento. Además, se muestran los programas empleados y desarrollados en el proyecto. Finalmente, se detallan las limitaciones que tiene el equipo y se concluye sobre los resultados del proyecto.

2. OBJETIVO

Explorar el uso de la cámara PixyCam para la detección de objetos con el fin de evaluar su uso en aplicaciones de monitoreo del tráfico.

3. ARDUINO

Arduino es una plataforma de programación de código abierto que funciona en conjunto con hardware de fácil uso. Las placas Arduino son capaces de leer entradas digitales de muchos accesorios y sensores, también tienen la capacidad de procesar estas entradas y generar salidas digitales que representen resultados.

Además, las placas Arduino tienen la capacidad de realizar tareas y tomar datos, gracias a las instrucciones generadas a través de la plataforma de Arduino para programación y almacenadas en la placa misma. Dicha interfaz de programación se presenta de la siguiente manera:

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

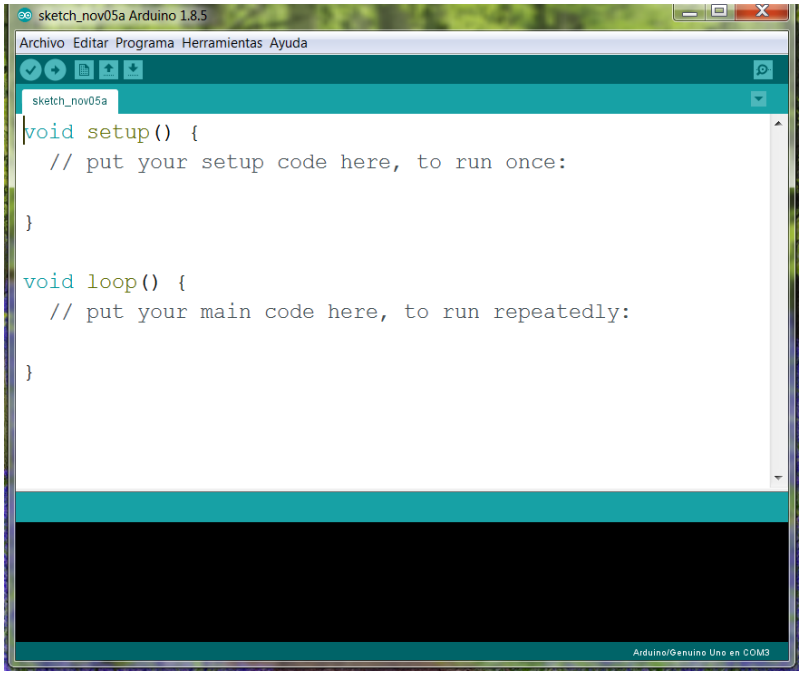


Figura 1. Interfaz de programación Arduino
Fuente: LanammeUCR, 2019

Una vez programada la serie de instrucciones deseadas, el archivo se guarda y luego se sube a la placa Arduino correspondiente con la ayuda de un cable de comunicación y suministro de energía que se conecta desde la computadora mediante USB hasta la placa (ver Figura 2).

Figura 2. Cable USB (comunicación/alimentación) placa Arduino
Fuente: Arduino, 2019

Una vez realizado este proceso, la placa Arduino queda lista para ejecutar las instrucciones ininterrumpidamente, en el caso de que las instrucciones incluyan la opción de desplegar resultados, estos se pueden observar al activar la ventana de Monitor Serie en la esquina superior derecha de la interfaz de programación como se observa en la Figura 3.

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

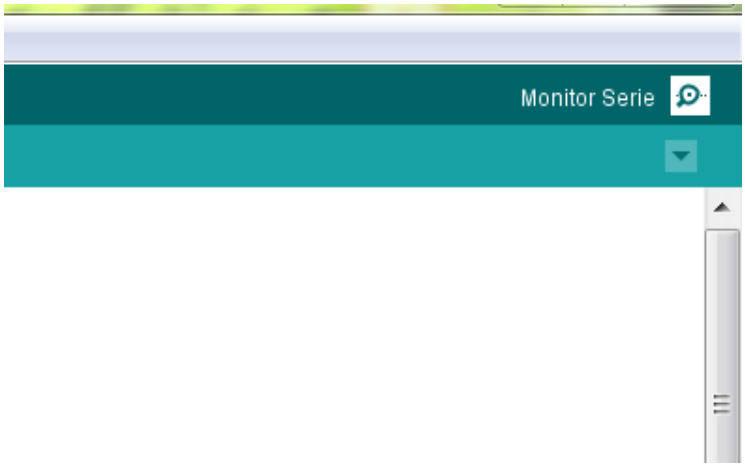


Figura 3. Botón de activación del Monitor Serie de Arduino
Fuente: LanammeUCR, 2019

Actualmente la Unidad de Seguridad Vial y Transporte (USVT) tiene a disposición una placa Arduino Uno, como la que se observa en la Figura 4. Esta placa ha sido la empleada en todas las fases del proyecto.

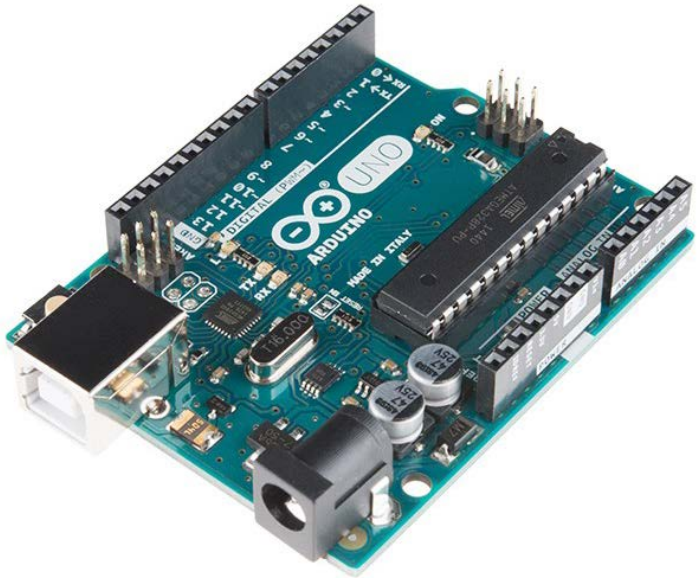


Figura 4. Placa Arduino Uno
Fuente: Arduino, 2019

Como antecedente práctico en el uso del Arduino se utilizó en el estudio de la medición del desplazamiento lateral de vehículos utilizando sensores ultrasónicos (Oconitrillo Varela, 2018) que fue un proyecto apoyado por la USVT.

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

4. PIXYCAM

Pixy es un sensor de visión rápida para proyectos sencillos de robótica y aplicaciones similares. A Pixy se le puede enseñar un objeto colorido para que aprenda a reconocerlo, además, tiene la capacidad de reconocer cientos de objetos a la vez. A este sensor también se le conoce usualmente como PixyCam, la Unidad de Seguridad Vial y Transporte (USVT) tiene a disposición dos de estos sensores, que se ven como el que se muestra en la Figura 5.

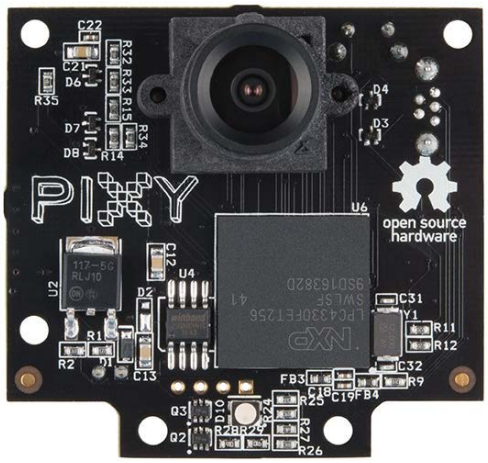


Figura 5. Accesorio PixyCam
Fuente: Pixy, 2019

Este dispositivo se conecta a una computadora mediante un cable USB de comunicación que tiene doble función de alimentar energía (ver Figura 6). Dado que el dispositivo no puede auto soportarse, se puede montar sobre una base denominada PixyCam Pan/Tilt (ver Figura 7).

Figura 6. Cable USB (comunicación/alimentación) PixyCam
Fuente: Pixy, 2019

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

Figura 7. Ensamble PixyCam Pan/Tilt
Fuente: Pixy, 2019

Con esta configuración sencilla, la PixyCam es capaz de detectar objetos mediante su interfaz de uso llamada PixyMon. Una vez conectado el dispositivo a la computadora e iniciada la aplicación PixyMon, se tiene lo observado en la Figura 8.

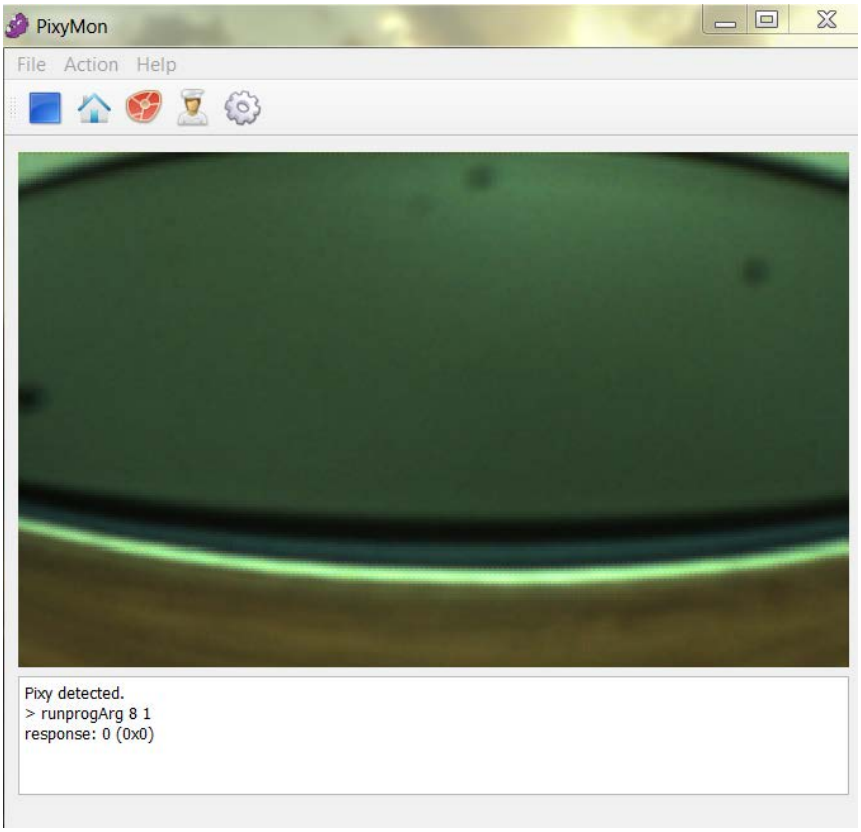


Figura 8. Interfaz PixyMon con PixyCam conectada
Fuente: LanammeUCR, 2019

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

Para realizar una asignación de objeto dentro de la memoria del dispositivo, con la intención de que él mismo lo identifique, se sigue la siguiente serie de pasos:

Paso 1

Se despliega el menú de Action, donde se selecciona la asignatura que se le quiere dar al objeto, en la Figura 9 se observa cómo se selecciona la asignatura 1 para asignar al objeto en pantalla.

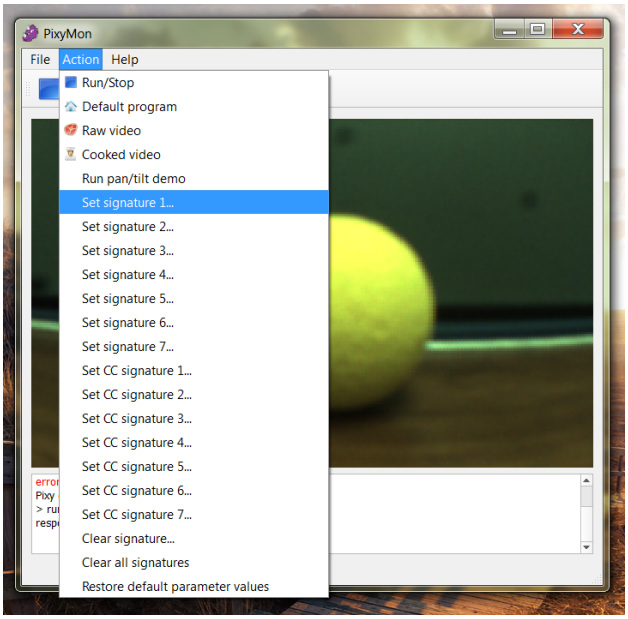


Figura 9. Selección de la asignatura 1 para el objeto en pantalla
Fuente: LanammeUCR, 2019

Paso 2

El objeto en pantalla se encierra con el uso del mouse haciendo uno cuadro que esté inscrito por completo dentro del objeto, como se observa en la Figura 10.

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

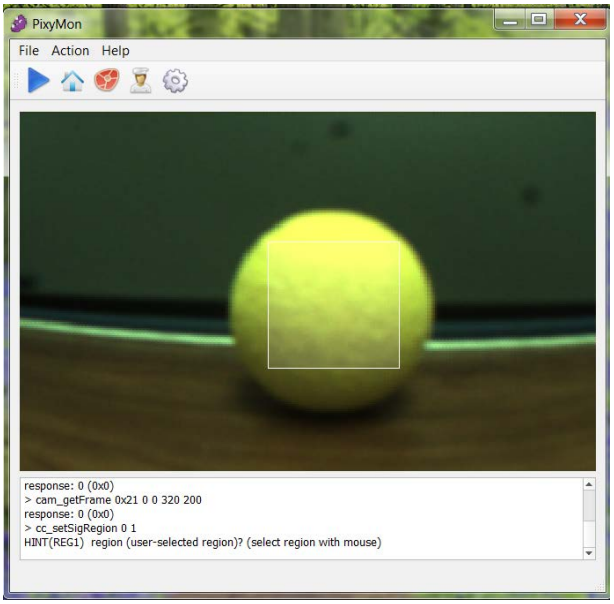


Figura 10. Selección objeto de pantalla
Fuente: LanammeUCR, 2019

Una vez realizada la selección del objeto, en pantalla se verá el objeto seleccionado encerrado en un cuadro, con pixeles asignados a la totalidad de su superficie y con su nombre de asignación en la esquina superior izquierda, como se muestra en la Figura 11.

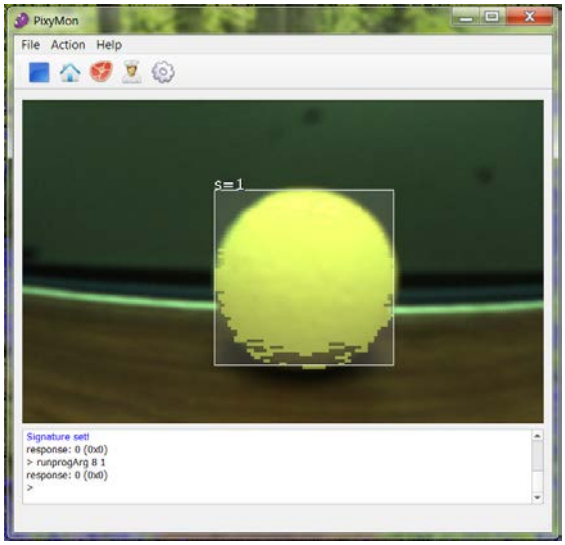


Figura 11. Objeto seleccionado con su respectivo nombre de asignación
Fuente: LanammeUCR, 2019

Paso 3

Finalmente se activa la pantalla del Default Program del PixyMon, dando click sobre el ícono en la esquina superior izquierda de la ventana, como se observa en la Figura 12. Esto despliega la interfaz de procesamiento de la cámara y es una vía de verificación para

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

corroborar que los objetos asignados se estén procesando de manera adecuada en pantalla. Esta interfaz también es la que se debe activar para lograr que el Arduino ejecute su programación y cumpla con sus funciones programadas.

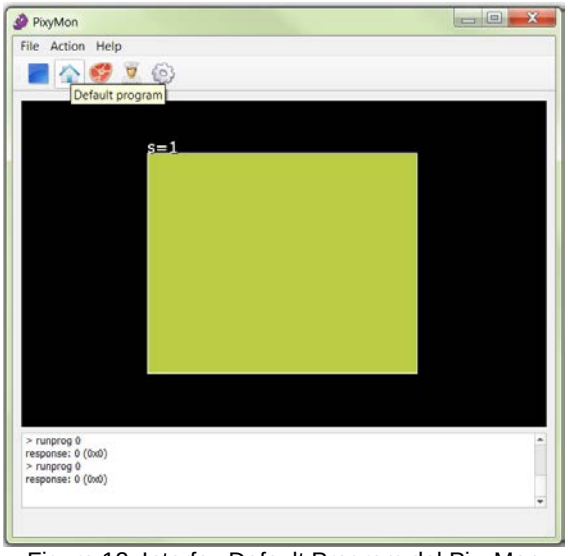


Figura 12. Interfaz Default Program del PixyMon
Fuente: LanammeUCR, 2019

La configuración completa para el procesamiento de datos y ejecución de la programación realizada en la interfaz Arduino, consiste en conectar la PixyCam al Arduino Uno, esto se realiza por medio de un cable tipo faja. En la Figura 13 se muestra la configuración. Además, ambos dispositivos se pueden conectar a la computadora, con la intención de verificar la salida del proceso en la interfaz de Arduino y también poder ver el registro de objetos a través del Default Program en el PixyMon.

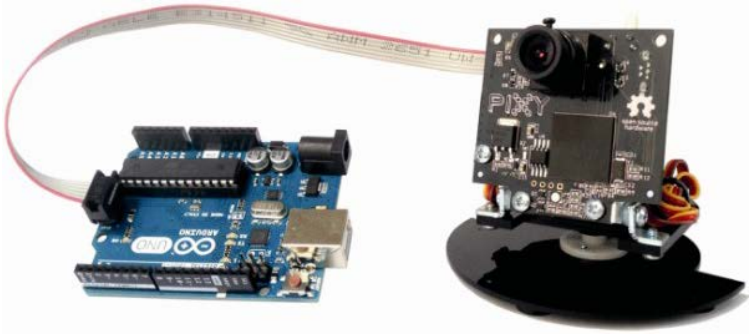


Figura 13. Configuración PixyCam Pan/Tilt conectada al Arduino Uno
Fuente: Arduino, 2019

5. PROGRAMAS UTILIZADOS Y DESARROLLADOS

A continuación, se enlistan los programas ejemplo de Arduino utilizados para diversas pruebas y los programas desarrollados para el procesamiento de los objetos detectados

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

por medio de la PixyCam. En el anexo se encuentra el código respectivo para cada programa.

5.1. Programa LED Cycle

El programa LED Cycle se incluye dentro de los ejemplos de programas de la interfaz de programación Arduino, este programa hace que el LED RGB de la placa Arduino Uno realice un ciclo de encendido/apagado entre sus colores. Se utilizó con la única finalidad de verificar la conexión adecuada entre la placa y la computadora. Al guardar el programa en la placa desde la interfaz Arduino, se verifica el completado del ciclo en la placa y se garantiza una correcta conexión.

5.2. Programa Bloques

Este programa fue el primero en desarrollarse desde la interfaz Arduino para la detección de objetos con la PixyCam, en síntesis, el programa despliega el número de objetos detectados en pantalla, su posición en los ejes “x” y “y”, junto con su ancho y alto.

Por las capacidades limitadas de la placa Arduino, la verificación de objetos en pantalla se realiza en intervalos de 1 segundo, este es el límite de procesamiento de la placa Arduino, exigir una mayor capacidad de procesamiento puede generar fallas temporales y/o permanentes en el funcionamiento de la placa.

El resultado obtenido y visto desde la interfaz Arduino a través del Monitor Serie es el siguiente:

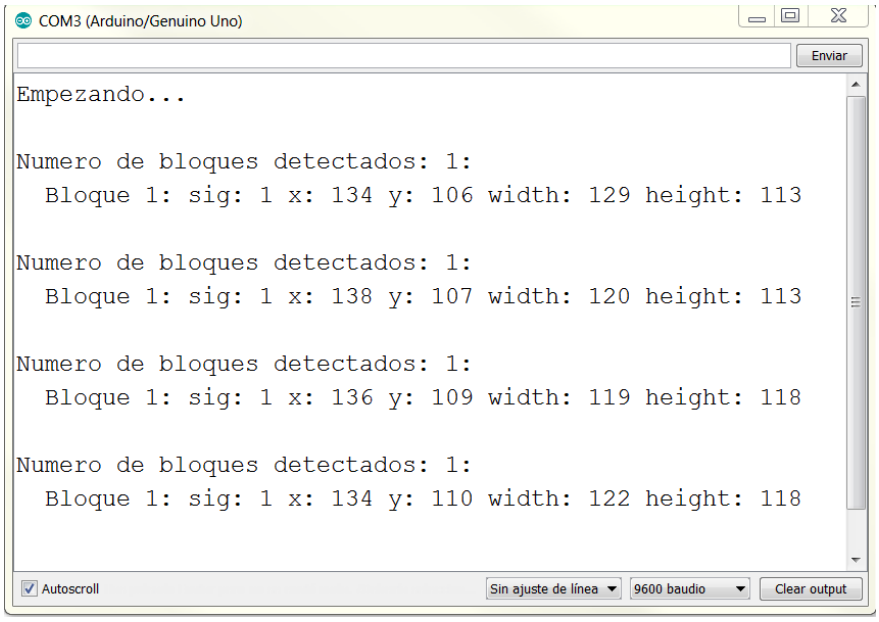


Figura 14.Resultado del programa Bloques en el Monitor Serie

Fuente: LanammeUCR, 2019

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

5.3. Programa PanTilt

El programa PanTilt se incluye dentro de los ejemplos de programa de la interfaz de programación Arduino que provee el creador del dispositivo PixyCam. Este programa está orientado específicamente para el montaje de la cámara sobre el PanTilt, que le permite a la cámara tener una base para auto soportarse y además le permite girar en direcciones gracias a un pequeño motor y engranajes.

El programa entonces le indica a la cámara que gire dentro de su rango disponible de giro mientras sigue el movimiento de un objeto almacenado en su memoria. Entonces cualquier objeto, que haya sido previamente almacenado en memoria, que entre en el campo de visión del dispositivo y se mueva, será seguido por la cámara hasta el punto en donde se salga de su rango de visión o hasta donde el radio de giro del mecanismo se lo permita.

Este programa cumplió un propósito de prueba dentro del desarrollo del proyecto, dado que se analizó su utilidad para la detección de objetos, sin embargo, se concluyó que, aunque el PanTilt ofrece una estructura de soporte a la cámara, lo mejor es que la misma se mantenga en una misma posición mientras realiza la detección de los objetos que se mueven frente a ella.

5.4. Programa Contador

Este fue el primer intento de programación para lograr que el arreglo propuesto realizara el conteo de bloques en pantalla. En resumen, el programa le solicita a la cámara verificar lo que tiene al frente, en caso de que tenga un solo objeto frente a ella, lo cuenta, luego un segundo después repite el proceso. Si entre ciclos un mismo objeto permanece en pantalla, el programa tiene la capacidad de notarlo y no contarlo por segunda vez, esto gracias a la medición de su altura y su ancho, bajo una tolerancia que se puede modificar dentro del código.

En caso de que haya más de un objeto en pantalla, el programa tiene la capacidad de diferenciar entre objetos, de nuevo, gracias a sus dimensiones y una tolerancia. Entonces, se suma al conteo la cantidad de objetos en pantalla y se hace un esfuerzo por poder diferenciar entre un objeto que permanece en pantalla y otro que entra por primera vez.

El resultado obtenido y visto desde la interfaz Arduino a través del Monitor Serie es el siguiente:

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

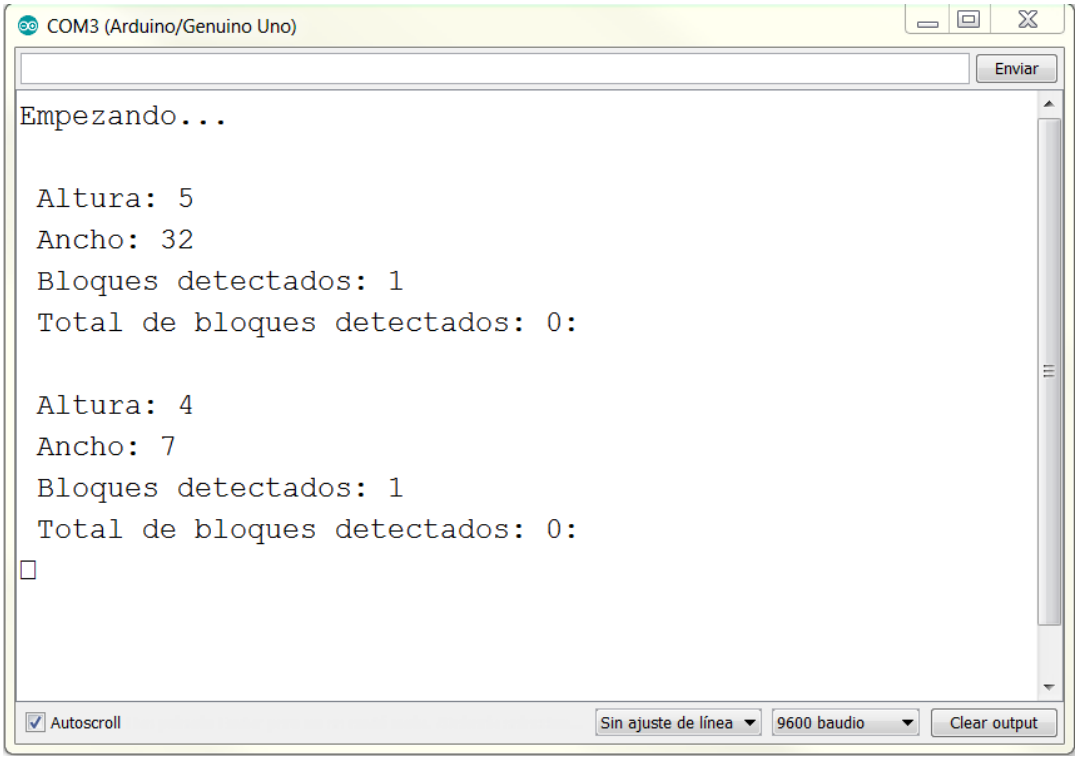


Figura 15. Resultado del programa Contador en el Monitor Serie
Fuente: LanammeUCR, 2019

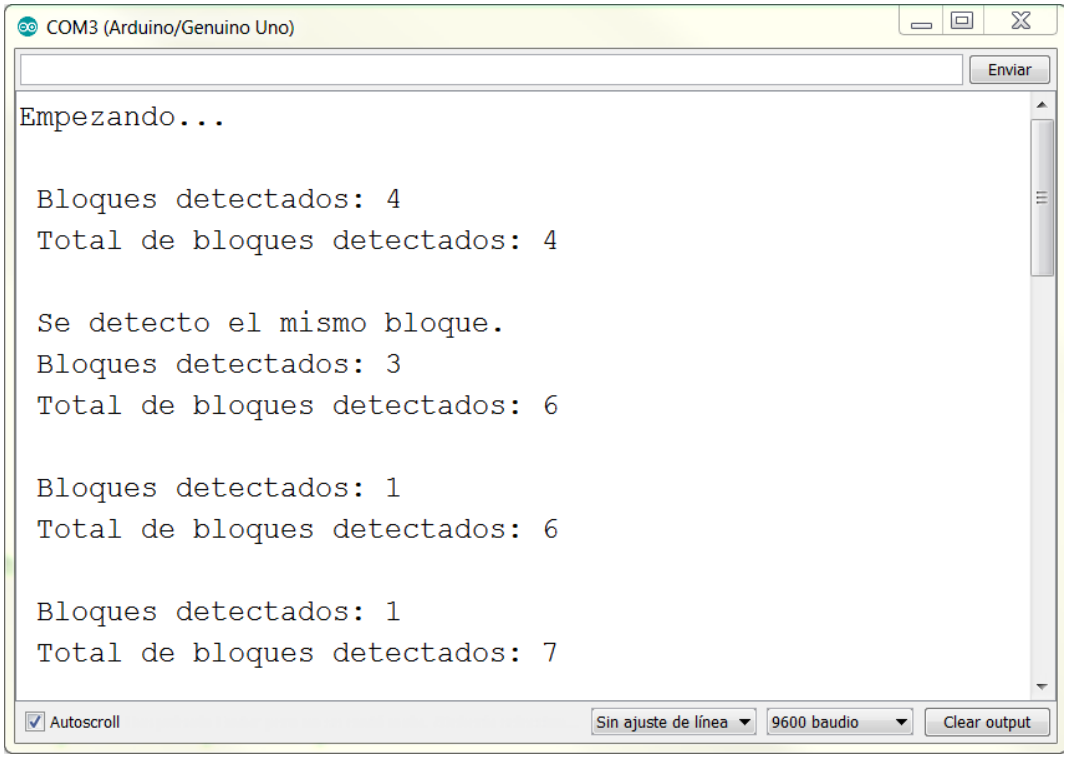
5.5. Programa Contador Cuatro Variables

Este programa es el segundo enfoque al problema de contar objetos en pantalla con el uso del arreglo de hardware propuesto. Tiene una base de funcionamiento parecida al programa Contador, con la gran diferencia de que contempla casos específicos. En primera instancia se tiene un caso específico para cuando hay un solo objeto en pantalla, a dicho objeto se le asignan cuatro variables para sus dimensiones, que luego se comparan cada segundo para determinar si el mismo objeto aún permanece en pantalla o hay uno nuevo; y entonces agregar un objeto al conteo total.

Luego, en casos donde se tenga más de un objeto en pantalla, se trabaja con vectores para asignar sus dimensiones en casillas del vector y poder comparar entre objetos, en dado caso que la diferencia entre dimensiones, registradas cada segundo, sea menor a la tolerancia, los bloques no se cuentan porque se concluye que han permanecido en pantalla. En caso que, sí haya un cambio significativo en las dimensiones, se asume que son bloques nuevos y se suman al total del conteo.

El resultado obtenido y visto desde la interfaz Arduino a través del Monitor Serie es el siguiente:

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.



Resultado del programa Contador 4 Variables en el Monitor Serie
Fuente: LanammeUCR, 2019

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

6. LIMITACIONES

Dentro de las principales limitaciones de cada uno de los equipos y software empleados se tiene:

- La placa Arduino Uno tiene únicamente la capacidad de procesar información en pantalla en un intervalo de un segundo, lo que se puede prestar para que un objeto que recorra una trayectoria frente a la cámara entre ese intervalo no sea detectado.
- La cámara PixyCam es capaz de reconocer solo objetos de ciertos colores y con cierta intensidad o brillo. Por lo que no se considera adecuado para la detección y monitoreo de vehículos o personas.
- inestabilidad en la detección de los objetos por parte de la PixyCam, esto es debido a la baja resolución provoca que la dimensión y posición aparente del objeto en pantalla constantemente este cambiando, aunque el objeto permanezca quieto.
- Debido al cambio de posición y dimensión aparente de los objetos en pantalla, la programación del Arduino para distinguir entre objetos nuevos en pantalla y permanecientes en la misma, es muy dificultosa.
- Todo objeto que quiera ser detectado con el uso de la PixyCam debe de ser previamente almacenado en la memoria del dispositivo para que sea debidamente reconocido cuando se presente ante el mismo.

7. CONCLUSIONES

Al finalizar el análisis exploratorio de detección de objetos con el arreglo de equipo Arduino Uno más PixyCam se concluye lo siguiente:

- El Arduino Uno es una placa electrónica de gran versatilidad, que le permite realizar gran cantidad de tareas mediante una programación básica y con sensores adjuntos. Esta versatilidad, sin embargo, le hace carecer de funciones específicas para tareas especializadas, como la detección de objetos en general.
- El Arduino Uno y la PixyCam son dispositivos que cumplen funciones educativas en las primeras etapas de la programación, electrónica y la robótica. Sus destrezas son limitadas y se pueden considerar dispositivos útiles para proyectos de tipo educativo o para proyectos específicos con una limitada capacidad de procesamiento.

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

- La PixyCam no es el dispositivo adecuado para la detección de objetos de gran tamaño, como por ejemplo peatones, ciclistas o vehículos. Sus limitaciones de detección por color y su falta de resolución son barreras difíciles de romper por medio de programación a través de la interfaz de Arduino.
- El Arduino Uno puede proveer un camino para las etapas iniciales de prueba de algunos proyectos o bien, fungir como base para proyectos menores de mediciones simples, siempre teniendo en consideración cuales serían los sensores adecuados para la toma de datos.
- El programa contador fue un primer paso en el conteo de objetos, sin embargo, no satisface los requerimientos propuestos, dado que no es lo suficientemente preciso para diferenciar entre objetos que permanecían en pantalla y aquellos que entraban en el rango de visión de la cámara por primera vez.
- El programa Contador Cuatro Variables logró solucionar algunos problemas en cuanto a la detección de objetos que permanecían en pantalla. Sin embargo, la resolución de la cámara al detectar las dimensiones del objeto es tan baja, que, al intercambiar un objeto por otro, de dimensiones similares, pero no idénticas, la cámara no es capaz de diferenciarlos, entonces no suma al conteo total. En otras palabras, los bloques tienen que ser muy diferentes en dimensión y color para que la combinación entre el programa y el arreglo de hardware sea capaz de contarlos como objetos diferentes.

8. REFERENCIAS

Arduino. (2019). *¿Qué es Arduino?* Recuperado en Novimebre de 2019, de Introducción: https://www.arduino.cc/en/Guide/Introduction

Oconitrillo Varela, E. (2018). *Uso de sensores ultrasónicos en la medición del desplazamiento lateral vehicular en diferentes secciones de la Red Vial Nacional Primaria de Costa Rica.* Montes de Oca: Universidad de Costa Rica.

Pixy. (2019). *Pixy.* Recuperado el Noviembre de 2019, de https://pixycam.com/pixy-cmucam5/

Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

9. ANEXOS

9.1. Anexo 1 Código Programa LED Cycle

El código ejemplo correspondiente al programa es el siguiente:

```
// Inicio del Código
#include <SPI.h>
#include <Pixy.h>
Pixy pixy;
void setup()
{
  Serial.begin(9600);
  Serial.print("Starting...\n");
  pixy.init();
}
void loop()
{
   uint32_t i=0;
   uint8_t r, g, b;
   while(1)
   {
     // calculate r, g, b such that it cycles through the colors
     r = i&0xff;
     g = (i*3) & 0xff;
     b = (i/3) & 0xff;
     pixy.setLED(r, g, b);
     // We need to delay here because serial requests are handled
     // every frame period (20ms). If we don't delay, we'll
     // overrun Pixy's receive queue. But that's all OK because
     // we normally only update the LED once per frame anyway.
delay (20);
     i++;
   }
}
// Fin del Código
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

9.2. Anexo 2 Programa Bloques

El código ejemplo correspondiente al programa es el siguiente:

```
// Inicio del Código
#include <SPI.h>
#include <Pixy.h>
// This is the main Pixy object
Pixy pixy;
void setup()
{
  Serial.begin(9600);
Serial.print("Empezando...\n");
  pixy.init();
}
void loop()
{
  static int i = 0;
  int j;
  uint16_t blocks;
  char buf[32];
  // grab blocks!
  blocks = pixy.getBlocks();
  // If there are blocks, print them!
  if (blocks)
  {
    i++;
    // do this (print) every 50 frames because printing every
    // frame would bog down the Arduino
if (i%50==0)
    {
      Serial.print ("\n");
      sprintf(buf, "Numero de bloques detectados: %d:\n", blocks);
Serial.print(buf);
      for (j=0; j<blocks; j++)
      {
        sprintf(buf, "  Bloque %d: ", j+1);
        Serial.print(buf);
pixy.blocks [j].print();
      }
    }
  }
}
// Fin del Código
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

9.3. Anexo 3 Programa PanTilt

El código ejemplo correspondiente al programa es el siguiente:

```
// Inicio del Código
#include <SPI.h>
#include <Pixy.h>
Pixy pixy;
#define X_CENTER        ((PIXY_MAX_X-PIXY_MIN_X)/2)
#define Y_CENTER        ((PIXY_MAX_Y-PIXY_MIN_Y)/2)
class ServoLoop
{
public:
  ServoLoop(int32_t pgain, int32_t dgain);
  void update(int32_t error);
  int32_t m_pos;
  int32_t m_prevError;
  int32_t m_pgain;
  int32_t m_dgain;
};
ServoLoop panLoop(300, 500);
ServoLoop tiltLoop(500, 700);
ServoLoop::ServoLoop(int32_t pgain, int32_t dgain)
{
  m_pos = PIXY_RCS_CENTER_POS;
  m_pgain = pgain;
  m_dgain = dgain;
  m_prevError = 0x80000000L;
}
void ServoLoop::update(int32_t error)
{
  long int vel;
  char buf[32];
  if (m_prevError!=0x80000000)
  {
    vel = (error*m_pgain + (error - m_prevError)*m_dgain)>>10;
    //sprintf(buf, "%ld\n", vel);
    //Serial.print(buf);
    m_pos += vel;
    if (m_pos>PIXY_RCS_MAX_POS)
      m_pos = PIXY_RCS_MAX_POS;
    else if (m_pos<PIXY_RCS_MIN_POS)
      m_pos = PIXY_RCS_MIN_POS;
  }
  m_prevError = error;
}
void setup()
{
  Serial.begin(9600);
  Serial.print("Starting...\n");
  pixy.init();
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

```
}
void loop()
{
  static int i = 0;
  int j;
  uint16_t blocks;
  char buf[32];
  int32_t panError, tiltError;
  blocks = pixy.getBlocks();
  if (blocks)
  {
    panError = X_CENTER-pixy.blocks[0].x;
    tiltError = pixy.blocks[0].y-Y_CENTER;
    panLoop.update(panError);
    tiltLoop.update(tiltError);
    pixy.setServos(panLoop.m_pos, tiltLoop.m_pos);
    i++;
    // do this (print) every 50 frames because printing every
    // frame would bog down the Arduino
    if (i%50==0)
    {
      sprintf(buf, "Detected %d:\n", blocks);
      Serial.print(buf);
      for (j=0; j<blocks; j++)
      {
        sprintf(buf, "  block %d: ", j+1);
        Serial.print(buf);
        pixy.blocks[j].print();
      }
    }
  }
}
// Fin del Código
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

9.4. Anexo 4 Programa Contador

El código desarrollado correspondiente al programa es el siguiente:

```
// Inicio del Código
#include <SPI.h>
#include <Pixy.h>
// Principal Objeto de Pixy
Pixy pixy;
void setup()
{
  Serial.begin(9600);
  Serial.print("Empezando...\n");
  pixy.init();
}
void loop() //Ciclo infinito
{
  int count = 0;
  static int i = 0;
  int tam;
  int j = 0;
  int c = 0;
  int k = 0;
  int difh;
  int difw;
  uint16_t blocks;
  char buf[32];
  while (c==0) //Ciclo que se repita cada segundo
  {
  blocks = pixy.getBlocks(); // Obtiene los bloques
  // Si hay bloques.
  if (blocks)
  {
    i++;
    // Impresion cada 50 frames
    if (i%50==0)
    {
      int Alturas[]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
      int Anchos[]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
      if (blocks==1) //Cuando hay un solo bloque
      {
       //for (j; j=j+1; j++) //Llenando el vector de altura y ancho
        j++;
        Alturas[j]=pixy.blocks[j].height;
        Serial.print("\n Altura: ");
        Serial.print(Alturas[j]);
        Anchos[j]=pixy.blocks[j].width;
        Serial.print("\n Ancho: ");
        Serial.print(Anchos[j]);
       for (tam=j-1;tam<j;tam++) //Diferenciacion entre bloque
       {
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

```
      difh = abs(Alturas[tam]-Alturas[tam+1]);
      difw = abs(Anchos[tam]-Anchos[tam+1]);
      if(difh > 5)
      {
        count++;
        }
      }
     }
    //if (blocks>1)  //Cuando hay más de un bloque
    //{
     //count = count + blocks;
      //for (j=0; j<blocks; j++) //Diferenciacion entre bloques
      //pixy.blocks[j].height;
     //}
    Serial.print("\n Bloques detectados: ");
    Serial.print(blocks);
    sprintf(buf, "\n Total de bloques detectados: %d:\n", count);
    Serial.print(buf);
  }
 }
  }
}
// Fin del Código
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

9.5. Anexo 5 Programa Contador Cuatro Variables

El código desarrollado correspondiente al programa es el siguiente:

```
        // Inicio del Código
#include <SPI.h>
#include <Pixy.h>
Pixy pixy; // Principal Objeto de Pixy
void setup()
{
  Serial.begin(9600);
  Serial.print("Empezando...\n");
  pixy.init();
}
void loop() //ciclo que no avanza debido a que esta enciclado en el While
{
  int count=0;
  static int i = 0; int j;
  int h1; int h2=0; int a1; int a2=0; //Variables para el caso de que haya 1 bloque en pantalla
  int hi[5]; int ai[5];
  int hf[]={0,0,0,0,0}; int af[]={0,0,0,0,0}; //Vectores para el caso de mas de un bloque en pantalla
  int c = 0; int test=0; int prueba=0;
  bool x = true;
  uint16_t blocks;
  char buf[32];
  int tolerancia = 3;
  while (c==0) //Ciclo cada segundo
  {
  blocks = pixy.getBlocks(); // Obtiene los bloques
  if (blocks) // Si hay bloques.
   {
    i++;
    if (i%50==0)  // Impresion cada 50 frames para no sobreforzar el arduino
    {
      if (blocks==1) //Cuando hay un solo bloque
      {
        h1=pixy.blocks[0].height; //Altura y Ancho del bloque en pantalla
        a1=pixy.blocks[0].width;
        if (test==2) //Se viene de 2 o más bloques en pantalla a 1 solo
        {
        for(j=0;j<blocks;j++)
        {
          if (abs(hf[j]-h1)>=tolerancia && abs(af[j]-a1)>=tolerancia)
         {
           prueba=9; //Se hace un bypass para no contar un objeto extra
          }
         }
         }
        if(abs(h2-h1)>=tolerancia && abs(a2-a1)>=tolerancia && prueba!=9) //Diferenciacion entre 1
solo bloque en pantalla
        {
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

```
        count++; //Contador
        }
      if(abs(h2-h1)<=tolerancia && abs(a2-a1)<=tolerancia) //Verificacion de que el bloque sea el mismo
      {
        Serial.print("\n Se detecto el mismo bloque.");
        }
       h2=h1; //Se almacena valores de altura y ancho para comparar en el siguiente segundo
       a2=a1;
       test=1; //Se sale desde 1 solo bloque en pantalla
       prueba=0;
     }
     if (blocks>1) //Si hay mas de un bloque
     {
      if(test==1) //Si se viene de un solo bloque en pantalla
      {
        for(j=0;j<blocks;j++)
      {
        hf[j]=h2;
        af[j]=a2;
        }
        }
      for(j=0;j<blocks;j++)
      {
        hi[j]=pixy.blocks[j].height; //Almacenaje de valores para cada bloque en pantalla
        ai[j]=pixy.blocks[j].width;
        }
      for(j=0;j<blocks;j++)
      {
       if (abs(hi[j]-hf[j])>=tolerancia && abs(ai[j]-af[j])>=tolerancia) //Comparación entre bloques en pantalla
       {
        count++; //Contador
        }
       else //(abs(hi[j]-hf[j])<=tolerancia && abs(ai[j]-af[j])<=tolerancia) Verificacion de que sean los mismos bloques
       {
        Serial.print("\n Se detectó el mismo bloque.");
        }
      }
      for(j=0;j<blocks;j++)
      {
        hf[j]=hi[j]; //Se almacenan valores para la comparación del siguiente segundo
         af[j]=ai[j];
        }
        test=2; //Se establece que en pantalla habia 1 o más bloques
      }
    Serial.print("\n Bloques detectados: "); //Impresion de bloques en pantalla
    Serial.print(blocks);
    sprintf(buf, "\n Total de bloques detectados: %d\n", cont.); //Impresión de bloques acumulados
    Serial.print(buf);
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.

```
    delay(1000);
    }
  }
 }
}
// Fin del Código
```


Tel.: +506 2511-2500 | Fax: +506 2511-4440 | direccion.lanamme@ucr.ac.cr | www.lanamme.ucr.ac.cr
Dirección: LanammeUCR, Ciudad de la Investigación, Universidad de Costa Rica.
Código Postal: 11501-2060, San Pedro de Montes de Oca, San José, Costa Rica.