

**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA**  
Facultad de Ingeniería, Arquitectura y Diseño  
**Programa de Maestría y Doctorado en Ciencias e Ingeniería**



---

**SISTEMA CONTADOR DE PECES EN TIEMPO REAL  
MEDIANTE PROCESAMIENTO DIGITAL DE IMÁGENES  
UTILIZANDO UN SISTEMA EMBEBIDO**

---

TESIS

que para cubrir parcialmente los requisitos necesarios para obtener el grado de

**DOCTOR EN CIENCIAS**

presenta:

**JUAN MIGUEL HERNÁNDEZ ONTIVEROS**

Director de tesis

Dr. Everardo Inzunza González

Ensenada, Baja California, México. Diciembre de 2017.

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Ingeniería, Arquitectura y Diseño

SISTEMA CONTADOR DE PECES EN TIEMPO REAL  
MEDIANTE PROCESAMIENTO DIGITAL DE IMÁGENES  
UTILIZANDO UN SISTEMA EMBEBIDO

TESIS

que para obtener el grado de DOCTOR en CIENCIAS presenta:

JUAN MIGUEL HERNÁNDEZ ONTIVEROS

Y aprobada por el siguiente comité:



---

Dr. Everardo Inzunza González

*Director del Comité*



---

Dr. Enrique Efraim García Guerrero

*Co-director del Comité*



---

Dr. Oscar Roberto López Bonilla

*Miembro del Comité*



---

Dr. Juan Iván Nieto Hipólito

*Miembro de Comité*



---

Dr. Marcial Castro Muñoz

*Miembro del Comité*

Agosto de 2017

**RESUMEN** de la tesis de **Juan Miguel Hernández Ontiveros**, presentada como requisito parcial para obtener el grado de DOCTOR EN CIENCIAS EN ELÉCTRICA, del programa de Maestría y Doctorado en Ciencias e Ingeniería de la UABC. Ensenada, B. C. México, Agosto de 2017.

## **SISTEMA CONTADOR DE PECES EN TIEMPO REAL MEDIANTE PROCESAMIENTO DIGITAL DE IMÁGENES UTILIZANDO UN SISTEMA EMBEBIDO**

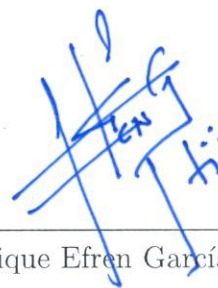
Resumen aprobado por:



---

Dr. Everardo Inzunza González

*Director de Tesis*



---

Dr. Enrique Efrén García Guerrero

*Co-director de Tesis*


El presente trabajo de tesis doctoral propone el desarrollo e implementación de un sistema para el conteo automático de peces vivos de tipo ornamental mediante el uso de un sistema embebido de alto rendimiento computacional y bajo costo. El sistema propuesto se prueba con 2 especies marinas: Guppies (*Poecilia reticulata*) y Mollies (*Poecilia sphenops*), bajo condiciones de iluminación controlada y con especímenes cuyas tallas varían de 0.5 a 2.3 cm. La originalidad de este trabajo de tesis versa en el estudio empírico para determinar el color de fondo del acuario dando como mejor resultado el amarillo, así como la integración e implementación del algoritmo propuesto en un sistema embebido de bajo costo y alto rendimiento computacional, específicamente Raspberry Pi 2, ejecutando el software de programación científica GNU Octave, lo que permite que el sistema de conteo sea económico, confiable, portable y escalable tecnológicamente entre diferentes sistemas operativos. Se hace un análisis de la tasa de exactitud AR (Accuracy Rate) y de su desviación estándar respecto a la cantidad real de peces, usando los mismos parámetros de trabajos reportados en la literatura, logrando con ello la comparación del desempeño entre una computadora personal y el sistema embebido propuesto, dando como principal resultado una exactitud promedio de 96.64% bajo condiciones de operación en tiempo real. El sistema propuesto tiene una mejor exactitud por más de 2.1% comparándolo contra los resultados obtenidos en los métodos actuales.

**Palabras clave:** Contador de peces, peces ornamentales, procesamiento digital de imágenes, sistema embebido, sistema en un chip.

**ABSTRACT** of the thesis of **JUAN MIGUEL HERNÁNDEZ ONTIVEROS**, presented as a partial requirement to obtain the degree of **DOCTOR** in **ELECTRICAL SCIENCES**, of the program of MSc and PhD in Sciences and Engineering of UABC. Ensenada, B. C., Mexico, August 2017.


## **REAL TIME FISH COUNTER SYSTEM BY DIGITAL IMAGE PROCESSING BY USING AN EMBEDDED SYSTEM**

Abstract approved by:



---

Dr. Everardo Inzunza González  
*Thesis supervisor*



---

Dr. Enrique Efrén García Guerrero  
*Thesis supervisor*

This doctoral thesis proposes the development and implementation of a system for automatic counting of live fish for ornamental type by using a high computational performance and low cost embedded system. The proposed system is tested with 2 marine species: Guppies (*Poecilia reticulata*) and Mollies (*Poecilia sphenops*), under controlled lighting conditions and with specimens whose sizes vary from 0.5 to 2.3 cm. The originality of this thesis work lies in the empirical study to determine the background color of the aquarium, which is the yellow color, as well as the integration and implementation of the algorithm proposed in an embedded system of low cost and high computational performance, specifically Raspberry Pi 2, running the GNU Octave scientific programming software, which allows the counting system to be economical, reliable, portable, and technologically scalable among different operating systems. An analysis of the Accuracy Rate (AR) and its standard deviation with respect to the actual quantity of fish is done by using the same work parameters reported in the literature, thus achieving a comparison of the performance between a personal computer and the proposed embedded system, obtaining as its main result an average accuracy of 96.64% under real-time operating conditions. The proposed system has a better accuracy of more than 2.1% comparing it against the results obtained in the current methods.

**Keywords:** Fish counter, ornamental fish, digital image processing, embedded system, system on a chip.

*A mi amada esposa Carolina*

# Agradecimientos

*Al Dr. Everardo Inzunza González, quien se ganó mi admiración y respeto por su gran calidad humana, por esmerarse tanto a la hora de impartir sus cátedras y por haber sido más que un director de tesis un amigo en quien pude confiar tanto en el ámbito académico como en lo personal.*

*Al Dr. Enrique Efren García Guerrero, por sus comentarios, correcciones y sugerencias durante el desarrollo de la tesis, y por las pláticas tan amenas que tuve en su cubículo en las cuales me orientaba sobre como realizar mejor el trabajo de investigación.*

*Al Dr. Oscar Roberto López Bonilla, por la ayuda que me brindó durante el desarrollo de la tesis, y por todas sus observaciones y sugerencias que realmente tomé en cuenta y me sirvieron para realizar esta tesis.*

*Al Dr. Juan Iván Nieto Hipólito, por sus consejos y sugerencias dadas en clases y durante la realización de este trabajo de tesis.*

*Al Dr. Marcial Castro Muñoz, por su amistad, su apoyo en los cursos y por sus atinados comentarios en los avances de tesis en donde daba un nuevo rumbo a la investigación con sus brillantes ideas.*

*A mi esposa Carolina Navarro Berrelleza, por ser el amor de mi vida, por tenerme paciencia durante las largas noches de estudio que duró esta aventura del doctorado y por hacerme sentir especial diariamente.*

*A mis padres Miguel Angel Hernández Aguirre y Silvia Ontiveros Valdés, por darme la vida y por estar en todo momento a mi lado.*

*Al CONACyT, que con su programa de becas me proporcionaron los recursos necesarios durante esta etapa importante en mi vida profesional.*

*Al PRODEP y al proyecto de investigación 485 aprobado en la 18ª convocatoria interna de UABC.*

*A nuestra alma mater: Universidad Autónoma de Baja California, que es como mi segundo hogar y donde se me ha permitido desarrollarme profesionalmente.*

Ensenada, B. C. México.  
Diciembre de 2017.

JUAN MIGUEL HERNÁNDEZ ONTIVEROS

# Tabla de Contenido

	Página
Tabla de Contenido	i
Resumen	iii
Abstract	iv
Agradecimientos	vi
Lista de figuras	ix
Lista de tablas	xii
<b>I Introducción</b>	<b>1</b>
I.1 Motivación . . . . .	1
I.2 Planteamiento del problema . . . . .	1
I.3 Propuesta de solución . . . . .	2
I.4 Objetivos . . . . .	3
I.4.1 Objetivo general . . . . .	3
I.4.2 Objetivos específicos . . . . .	3
I.5 Trabajos relacionados . . . . .	3
I.5.1 Trabajos relacionados en el ámbito científico . . . . .	3
I.5.2 Trabajos relacionados en el ámbito tecnológico . . . . .	6
I.5.3 Contadores de peces comerciales . . . . .	10
I.6 Organización del trabajo de tesis . . . . .	12
<b>II Marco Teórico</b>	<b>14</b>
II.1 Introducción . . . . .	14
II.2 Fundamentos de procesamiento digital de imágenes . . . . .	15
II.2.1 Histograma de imágenes . . . . .	18
II.3 Fundamento matemático . . . . .	19
II.4 Momentos invariantes de Hu . . . . .	21
II.5 Sistema embebido Raspberry Pi . . . . .	23
II.6 Especies marinas a contabilizar . . . . .	25
II.6.1 Guppies (Poecilia reticulata) . . . . .	25
II.6.2 Mollies (Poecilia sphenops) . . . . .	27
<b>III Desarrollo del sistema propuesto</b>	<b>28</b>
III.1 Construcción de las estructuras metálicas . . . . .	28
III.2 Adquisición de imágenes digitales . . . . .	33
III.2.1 Calibración del sistema propuesto . . . . .	37
III.3 Algoritmos propuestos . . . . .	38
III.3.1 Identificación y conteo utilizando momentos invariantes de Hu (Primer algoritmo) . . . . .	38
III.3.2 Implementación del segundo algoritmo propuesto en Matlab/GNU Octave . . . . .	41

# Tabla de Contenido (Continuación)

	Página
III.4 Integración del sistema propuesto . . . . .	48
III.5 Perspectiva de implementación en una granja de peces . . . . .	51
<b>IV Resultados experimentales</b>	<b>53</b>
IV.1 Resultados utilizando momentos invariantes de Hu (primer algoritmo propuesto) . . . . .	53
IV.2 Resultados del segundo algoritmo propuesto empleando una PC con Matlab . . . . .	60
IV.3 Resultados empleando Raspberry Pi con GNU Octave . . . . .	64
IV.4 Comparación de resultados entre ambas plataformas . . . . .	78
IV.5 Comparación de resultados con trabajos relacionados . . . . .	79
<b>V Conclusiones generales y aportaciones</b>	<b>81</b>
V.1 Conclusiones . . . . .	81
V.2 Aportaciones . . . . .	82
V.3 Trabajos a futuro . . . . .	83
<b>Bibliografía</b>	<b>85</b>
<b>A Dibujo y Perspectiva de la Estructura Mecánica</b>	<b>92</b>
<b>B Producción Académica</b>	<b>94</b>
<b>C Glosario de términos</b>	<b>96</b>

# Lista de figuras

Figura		Página
1	Contador manual. . . . .	10
2	Contador de células para microscopios de la marca comercial Celeromics. . . . .	11
3	Contador Faivre de especies marinas. . . . .	12
4	Contadores Vaki Micro Counter y Macro Counter. . . . .	13
5	Representación de una imagen en escala de grises. . . . .	15
6	Representación de una imagen a color en RGB. . . . .	17
7	Histograma de la imagen digital mostrada en la figura 5. . . . .	18
8	Raspberry Pi 2 Modelo B. . . . .	24
9	Memoria microSD clase 10 de 32 GB. . . . .	24
10	Cámara para Raspberry Pi 2 Modelo B de 5 MP. . . . .	24
11	Peces Guppies ( <i>Poecilia reticulata</i> ). . . . .	26
12	Pez Molly hembra ( <i>Poecilia sphenops</i> ). . . . .	27
13	Diagrama a bloques del primer sistema propuesto. . . . .	28
14	Primera estructura metálica del contador de peces propuesto. . . . .	29
15	Diagrama a bloques del segundo sistema propuesto. . . . .	30
16	Segunda estructura metálica del contador de peces propuesto. . . . .	31
17	Vista superior de la segunda estructura con la carcasa de lona amarilla protectora del exceso de luz exterior. . . . .	31
18	Acuario desarrollado para el sistema contador de peces. . . . .	32
19	Raspberry Pi montada en la estructura metálica. . . . .	32
20	Analogía de señal con rango de valores de pixeles. . . . .	36
21	Enfoque no deseado de la imagen de entrada. . . . .	37
22	Enfoque calibrado de la imagen capturada en donde solo aparece la base del acuario. . . . .	38
23	Diagrama de flujo del algoritmo basado en los momentos invariantes de Hu. . . . .	39
24	Plantilla base de peces a identificar. . . . .	40
25	Diagrama de flujo del segundo algoritmo propuesto. . . . .	43
26	Visualización del procesamiento de una imagen del acuario sin peces, con fondo color Amarillo Huevo. A) Imagen de entrada. B) Imagen binarizada. C) Imagen complementada. . . . .	45
27	Visualización del procesamiento de una imagen del acuario sin peces de fondo color blanco. A) Imagen de entrada. B) Imagen binarizada. C) Imagen complementada. . . . .	46
28	Arreglo experimental del primer sistema automático contador de peces basado en procesamiento digital de imágenes y empleando una computadora personal (PC). . . . .	49

# Lista de figuras (Continuación)

Figura	Página
29 Arreglo experimental del segundo sistema automático contador de peces basado en procesamiento digital de imágenes y empleando el sistema embebido Raspberry Pi 2. . . . .	50
30 Implementación del sistema en una granja de peces. . . . .	52
31 Imagen con 70 peces de 5 especies diferentes, ordenadas aleatoriamente y de diferentes tamaños. . . . .	54
32 A) Imagen original binarizada, B) Identificación de cada pez. . . . .	54
33 Ejemplo de extracción individual de los peces de la imagen original empleando momentos de Hu. . . . .	55
34 Resultado de la identificación y contabilización de peces. . . . .	55
35 Imagen original con 70 peces y con 4% de ruido del tipo sal y pimienta. . . . .	57
36 Imagen original con 70 peces y con 4% de ruido gaussiano, media 0 y varianza 0.04. . . . .	57
37 Clasificación en 3D de peces a identificar bajo ruido del tipo sal y pimienta. . . . .	58
38 Resultados del procesamiento digital de la imagen de entrada usando Matlab: A) Imagen de entrada con 70 peces, B) Imagen binaria, C) Identificación de los peces. . . . .	61
39 Situaciones aleatorias susceptibles a motivar errores en el conteo de peces. . . . .	63
40 Resultados del procesamiento digital de la imagen de entrada usando Raspberry Pi: A) Imagen de entrada con 130 peces, B) Imagen binaria, C) Identificación de los contornos de los peces. . . . .	65
41 Secuencia de 16 imágenes capturadas con cámara de Raspberry Pi cada 2 segundos a la familia de Guppies ( <i>Poecilia reticulata</i> ) con una talla promedio de 1.2 cm. . . . .	66
42 Resultados de contabilizar 350 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 1.2 cm. . . . .	67
43 Resultados de contabilizar 250 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 1.2 cm. . . . .	68
44 Resultados de contabilizar 150 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 1.2 cm. . . . .	69
45 Resultados de contabilizar 65 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 1.2 cm. . . . .	70
46 Resultados de contabilizar 195 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 2.3 cm. . . . .	71
47 Resultados de contabilizar 115 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 2.3 cm. . . . .	72
48 Resultados de contabilizar 65 Guppies ( <i>Poecilia Reticulata</i> ) con una talla promedio de 2.3 cm. . . . .	73

# Lista de figuras (Continuación)

Figura		Página
49	Resultados de contabilizar 130 Mollies (Poecilia Sphenops) con una talla promedio de 0.5 cm. . . . .	75
50	Resultados de contabilizar 95 Mollies (Poecilia Sphenops) con una talla promedio de 0.5 cm. . . . .	75
51	Resultados de contabilizar 50 Mollies (Poecilia Sphenops) con una talla promedio de 0.5 cm. . . . .	76
52	Dibujo del contador de peces. . . . .	92
53	Perspectivas del contador de peces. . . . .	93

# Lista de tablas

Tabla	Página
I Valor promedio de pixeles de las imágenes de fondo del acuario obtenidos con una PC usando Matlab. . . . .	34
II Valor promedio de pixeles de las imágenes de fondo del acuario obtenidos con Raspberry Pi 2 y GNU Octave. . . . .	35
III Calibración del algoritmo utilizado en el primer sistema basado en PC con Matlab: Se procesan y promedian 16 imágenes sin peces para cada caso con diferentes colores de fondo y distintas condiciones de iluminación.	47
IV Calibración del algoritmo utilizado en el segundo sistema basado en Raspberry Pi con GNU Octave: Se procesan y promedian 16 imágenes sin peces para cada caso con diferentes colores de fondo y distintas condiciones de iluminación. . . . .	47
V Momentos de Hu de los peces a identificar. . . . .	56
VI Validación del primer algoritmo tomando en cuenta ruido sal y pimienta y ruido gaussiano. . . . .	59
VII Conteo de peces de la especie Guppies ( <i>Poecilia reticulata</i> ) con un tamaño de 5 mm usando el sistema contador automático basado en PC y Matlab (segundo algoritmo propuesto). . . . .	62
VIII Pruebas de conteo de diferentes especies de peces y diferentes tallas, a partir de una sola imagen capturada de peces vivos, en todos los casos la cantidad real de peces fue de 115 usando PC y Matlab. . . . .	64
IX Validación del conteo de peces de la especie Guppies ( <i>Poecilia reticulata</i> ) con un tamaño de 1.2 cm usando el sistema contador automático basado en Raspberry Pi. . . . .	70
X Conteo de peces de la especie Guppies ( <i>Poecilia reticulata</i> ) con un tamaño de 2.3 cm usando el sistema contador automático con Raspberry Pi. . . . .	74
XI Conteo de peces de la especie Mollies ( <i>Poecilia Sphenops</i> ) con tamaño de 0.5 cm usando el sistema contador automático con Raspberry Pi. . .	77
XII Costo en dólares de los sistemas usados para el conteo de peces, sistema embebido (Raspberry Pi) y PC (MacBook Pro). . . . .	78
XIII Pruebas de conteo de diferentes especies de peces y diferentes tallas obteniendo la desviación estándar y exactitud del instrumento implementado.	79
XIV Comparación de resultados con los métodos de clasificación del estado del arte. . . . .	80

# Capítulo I

## Introducción

### I.1 Motivación

La motivación principal de este trabajo de tesis es el poder resolver un problema real, en este caso en particular es el conteo de peces vivos. Realizar una estimación precisa y compleja de los peces dentro del mar o dentro de un acuario es una tarea importante en las ciencias del mar. Dentro del mar el método tradicional es mediante buzos realizar un conteo manual de los peces, por otro lado dentro de un estanque con agua, lo común es capturar una imagen y realizar en ella el conteo manualmente. Es notorio que ambos métodos tienen sus limitaciones, la principal es el error humano al momento de realizar el conteo ya sea dentro del mar o en el estanque con agua, estos errores pueden ser debido a la fatiga, a condiciones emocionales, limitaciones biológicas y/o factores ambientales. El problema del conteo de peces en tiempo real y de forma eficiente, es un tema novedoso, de relevancia internacional donde múltiples investigadores dan posibles soluciones a este problema. En este trabajo de tesis se aborda una solución original, en donde se involucra un sistema embebido de alto rendimiento computacional y el diseño de un algoritmo capaz de contar los peces en tiempo real, de forma eficiente, de bajo costo y limitando los posibles errores humanos.

### I.2 Planteamiento del problema

En este trabajo de tesis se plantea solucionar un problema real de conteo de peces vivos de tipo ornamental, debido a que resulta ser una necesidad básica y prioritaria

en los procesos de cultivos acuícolas, el contabilizar de manera rápida y confiable la cantidad de la especie marina en proceso de crecimiento en alguna de las etapas de su producción. Se hace necesario conocer la cantidad de la especie bajo cultivo, por varias razones: i) proveer las condiciones óptimas del entorno del cultivo para optimizar el crecimiento y sanidad de la especie, ii) mantener los inventarios de producción en los límites establecidos y iii) en el proceso propio de la comercialización de la especie.

El hecho de contabilizar una especie acuícola bajo cultivo, es un problema que resulta complejo por la gran cantidad de variables presentes: i) tipo de especie (talla, forma, opacidad, etc.), ii) la cantidad (movilidad, comportamiento grupal, etc.) y iii) condiciones del medio en que se encuentra el cultivo (granja al aire libre, espacio cerrado, agua turbia, niveles de iluminación, profundidad de los estanques, etc.).

Una práctica común e implementada en procesos acuícolas, es que a partir de una fotografía capturada a pequeñas cantidades de la especie, se realiza el conteo visual en forma manual determinando así la cantidad respectiva. Esta práctica convencional, propicia errores humanos que adicionan incertidumbre al conteo propio y es un proceso lento. Bajo éste contexto, se presenta un método para el conteo de peces vivos en tiempo real.

### **I.3 Propuesta de solución**

Las etapas de operación del dispositivo para efectuar el conteo automático son: i) captura la imagen con la distribución en movimiento de la especie de peces de interés depositados en la pecera, ii) la imagen se acondiciona a través de un pre-procesamiento digital, iii) la imagen acondicionada ingresa a un algoritmo que contabiliza los peces detectados y iv) se despliega el resultado de la cantidad de peces encontrados en tiempo real.

## I.4 Objetivos

### I.4.1 Objetivo general

Desarrollar e implementar un sistema autónomo de conteo eficiente en tiempo real empleando técnicas de procesamiento digital de imágenes, de peces vivos en procesos de cultivo acuícolas.

### I.4.2 Objetivos específicos

- Desarrollar un programa capaz de identificar objetos.
- Realizar un estudio del espectro visible para encontrar el color ideal del fondo del acuario con respecto a la sensibilidad de la cámara.
- Desarrollar un algoritmo capaz de contabilizar los peces de forma automática.
- Diseñar una estructura de metal que contenga los espacios necesarios para el sistema de iluminación, el sistema embebido y la pecera.
- Realizar conteos con diferentes especies de peces para verificar el correcto funcionamiento del sistema propuesto.
- Realizar un evaluación del sistema propuesto.

## I.5 Trabajos relacionados

### I.5.1 Trabajos relacionados en el ámbito científico

En las últimas tres décadas, se han reportado en la literatura algunos trabajos que muestran diferentes métodos y algoritmos para el conteo de peces. En (Newbury *et al.*, 1995), (Zheng y Zhang, 2010), (Fan y Liu, 2013), (Shen *et al.*, 2011) y (Song *et al.*, 2014) se plantea la aplicación de redes neuronales para contabilizar peces de forma automática teniendo una exactitud de entre 94% y 98% respectivamente. En (Karplus

*et al.*, 2005) se presenta la propuesta de una máquina para clasificar Guppies, éstos se hacen pasar a través de tubos transparentes empujados por un flujo y nivel de agua controlados. En (Fabic *et al.*, 2013) describen un método para la detección, conteo y clasificación de peces bajo el agua. En (Baker y Gruner, 2007) se desarrolla un prototipo que mide el flujo de peces el cual se basa en sensores electromagnéticos. En (Ferrero *et al.*, 2014) se diseñan dos métodos ópticos de bajo costo para contabilizar peces, el primero de ellos esta basado en un conjunto de emisores y receptores infrarrojos que detectan la presencia de un pez cuando el haz de luz se divide por la silueta de un pez, el segundo método usa dos cámaras Web, una tarjeta comercial de adquisición de datos y un programa para procesar las imágenes desarrollado en LabVIEW<sup>TM</sup>. En (Li *et al.*, 2015) encuentran peces en una imagen mediante el uso de la binarización y de los gradientes normados. En (Martínez, 2015), (Morais *et al.*, 2005), (Toh *et al.*, 2009) (Loh *et al.*, 2011) y (Labuguen *et al.*, 2012) se utilizan técnicas de visión por computadora para contar peces vivos bajo el agua.

En (Zion *et al.*, 1999) y (Zion *et al.*, 2000) se propone una máquina de visión y procesamiento de imágenes para la clasificación automática del tipo de especie de peces, utilizando los momentos invariantes de Hu. En (Zion *et al.*, 2007) se presenta el desarrollo de dos sistemas encaminados a la clasificación de peces, el primer sistema utiliza una fuente de iluminación de fondo formada de ocho luces fluorescentes, el segundo sistema emplea una cámara CCD (Charge-Coupled Device) monocromática instalada bajo el agua para la captura de las imágenes. En (Zion *et al.*, 2008) se propone un algoritmo que clasifica y cuenta Guppies (*Poecilia reticulata*) por género, haciendo uso de procesamiento de imágenes. En (Zion, 2012) se describe ampliamente el estado del arte y la evolución de los últimos 25 años sobre el uso de la tecnología de la visión por computadora como una herramienta indispensable en la acuicultura. En (Navarro, 2015) se diseña y construye un sistema de monitoreo remoto de temperatura,

oxígeno disuelto, pH, salinidad, conductividad eléctrica, total de sólidos disueltos, humedad relativa y nivel de líquido en estanques de abastecimiento dotado con alarmas preventivas vía GSM con ajuste en sus intervalos de operación, así mismo se contempla el diseño de un contador optoelectrónico de peces. En (Mondaca, 2017) se presenta el diseño e implementación de un contador optoelectrónico de peces.

Dado que este trabajo de tesis resuelve un problema de conteo de peces vivos de tipo ornamental, desarrollando un algoritmo dentro de un sistema embebido, particularmente se propone utilizar Raspberry Pi 2, la cual se aborda en el segundo capítulo, aunque en esta sección se hace mención de algunos trabajos que ya empiezan a utilizar esta tecnología para resolver problemas reales, tal es el caso de (Islam *et al.*, 2014), en donde se crea una máquina de votación electrónica basada en Raspberry Pi. Esta minicomputadora usa una cámara para capturar una imagen a cada ciudadano que quiere votar, posteriormente realiza un procesamiento de imágenes, identifica si el ciudadano es válido para votar y si aún no ha realizado ya su voto, finalmente permite realizar el voto al ciudadano una vez que haya pasado todos los filtros de control.

Por otra parte, en (Schumann-Bölsche y Schön, 2015) se estudian las oportunidades y desafíos de la logística humanitaria con una visión especial sobre las aplicaciones en redes de sensores dadas por Raspberry Pi. En (Lamine y Abid, 2014) y (Agrawal y Singhal, 2015) se usa Raspberry Pi para automatizar una casa. En (Amir *et al.*, 2014) se usa el procesamiento de imágenes para rastrear cables marinos mediante el uso de Raspberry Pi. En (Nguyen *et al.*, 2015) y (Nikhade, 2015) se diseña e implementa un sistema de monitoreo basado en Raspberry Pi.

Por otro lado, en (Sarango Chuquimarca, 2014) se presenta el diseño de un prototipo domótico, con el objetivo de medir en tiempo real el consumo de agua potable en una casa habitación, cuyo monitoreo se realiza por Internet y con reportes enviados al correo electrónico del usuario. El prototipo está compuesto por un sensor de flujo basado en el

efecto Hall, el sensor envía los datos medidos al núcleo principal del sistema, que es una Raspberry Pi. En (Sapes y Solsona, 2016) se presenta un sistema de reconocimiento de huellas digitales de bajo costo, embebido en Raspberry Pi con Linux. En (Leccese *et al.*, 2014) se presenta un estudio sobre la automatización de la iluminación del alumbrado público mediante el uso de paneles solares, Raspberry Pi, ZigBee y WiMAX.

Además, en (Catarino, 2015) se hace uso de Raspberry Pi en conjunto con Arduino para monitorear, registrar y visualizar diferentes variables eléctricas en diversos puntos de un proceso. Las variables que se monitorean dependen del proceso, las cuales pueden ser: temperatura, humedad, presión, concentración de gases, flujo, caudal, nivel de líquido, velocidad, peso, potencial de hidrógeno (pH), conductividad eléctrica, entre otras. En (Cristian Morelos Hernández, 2015) se diseña un sistema de seguridad basado en el Internet de las cosas y un microcontrolador con WiFi, dicho sistema incluye el envío de mensajes SMS (Short Message Service) con el apoyo de la plataforma IoT (Internet of Things) Carriots. En (Reyes Ibarra Luis Alberto, 2015) y (Aksentijević *et al.*, 2015) se usa en conjunto la tarjeta Raspberry Pi y el Internet de las cosas, en el primero de ellos se usa con el fin de controlar el riego en un invernadero y en el segundo se realiza una investigación que evalúa las posibilidades de la integración del Internet de las Cosas y la tarjeta Raspberry Pi.

### **I.5.2 Trabajos relacionados en el ámbito tecnológico**

La patente JP2008171196A registrada en Japón con fecha 11 de enero de 2007 por el inventor Maki Kazunari *et. al.*, se refiere a un sistema de conteo de peces que incluye una turbina hidráulica provista de una cámara y una computadora que analiza las imágenes tomadas por cada cámara (Maki Kazunari, 2007).

Las patentes US004034331 y US4439844 registradas en Estados Unidos con fecha 5 de Julio de 1977 y 27 de marzo de 1984 por los inventores Menin y Bridges, se refieren a

un contador de peces principalmente de salmones migratorios de corrientes, incluye un transductor de sonar de exploración lateral de extremo ancho. Esta estructura consiste de un marco ligero de tipo escalera con elementos transversales de soporte con el fin de evitar hundimientos. El transductor se suministra con pulsos transmitidos durante intervalos de corta duración (Menin, 1977) (Menin, 1984).

La patente US6809658 registrada en Estados Unidos con fecha 26 de octubre de 2004 por el inventor Lofaso, es un aparato electrónico contador de peces, el sistema incluye un contador electrónico, una placa frontal con una pantalla numérica LCD, un botón para incrementar o decrementar el conteo en la LCD, un botón de reinicio para restablecer la visualización numérica del LCD y una batería para llevar a cabo las funciones (Lofaso, 2004).

La patente US5321391 registrada en Estados Unidos con fecha 14 de junio de 1994 por el inventor Fox, propone un indicador de mordida de un pez, cuenta con una parte giratoria que esta enganchada por una línea de pesca, de manera que cuando esta en uso el movimiento longitudinal de la línea de pesca hace que la parte giratoria rote. Cuenta con un sensor de campo magnético el cual esta situado al interior del sistema de manera que cuando la parte giratoria se active provoca una alteración en el sensor de campo magnético generando una señal que indica mordedura de un pez (Fox, 1994).

La patente US4225951 registrada en Estados Unidos con fecha 30 de septiembre de 1980 por el inventor Menin y Fife, se refiere a un contador de peces con transductor de escaneo. Un sistema para contar peces en una maza de agua el cual utiliza uno o más transductores de lentes líquidos de enfoque acústico, cada uno de los cuales incluyen un gran número de elementos piezoeléctricos individuales dispuestos en arco para crear un patrón en forma de abanico (Menin, 1980).

La patente US08044337 registrada en Estados Unidos con fecha 25 de octubre de 2011 por el inventor Duszynski, propone un dispositivo para contar objetos de forma

electrónica y óptica, tales como peces, animales, vegetación u otros materiales, cuando son colocados dentro del recipiente. A medida que los objetos son colocados dentro del recipiente, caen a través del marco de la apertura e interrumpen un láser o LED generando el conteo (Duszynski, 2011).

Las patentes US5692064 y CA2134486C registradas en Estados Unidos con fecha 25 de noviembre de 1997 y 10 de noviembre de 1998 por los investores Takagi et. al., se refieren a un aparato que cuenta objetos bajo el agua usando ondas ultrasónicas. Se pueden contar un número de peces emitiendo una onda ultrasónica o una onda electromagnética hacia un área predeterminada por donde pasan los peces (por ejemplo un estanque de piscicultura), las ondas reflejadas por los peces se reciben desde una sección transversal del área y se almacenan periódicamente. Los peces se cuentan realizando un procesamiento de imagen que discrimina la imagen de sombra del pez de una onda ultrasónica basada en una regla de decisión. (Takagi Yoichi, 1997).

La patente US5260912 registrada en Estados Unidos con fecha 09 de noviembre de 1993 por el inventor Latham, se refiere a un aparato que busca peces de forma lateral el cual transmite dentro del agua un pulso de sonar y posteriormente monitorea los ecos del pulso. El intervalo de tiempo de los pulsos es aleatorio con el fin de permitir que dos o más unidades similares se utilicen estando ubicadas cerca pero sin interferirse entre ellas. Un eco individual se trata como la detección de un pez si tiene una magnitud por encima de un umbral predeterminado (Latham, 1993).

La patente US5637838 registrada en Estados Unidos con fecha 10 de junio de 1997 por el inventor Arey et. al., se refiere a un aparato que mide, pesa y contabiliza peces que comprenden una parte principal del cuerpo como los bordes paralelos laterales largos y cortos en una configuración rectangular, teniendo la porción de cuerpo principal una cara frontal, una cara posterior y una ranura para la recepción de un pez deslizándolo (Arey Clyde D., 1977).

La patente ES2280096 registrada en España con fecha 01 de septiembre de 2007 por el inventor Kubota, se refiere a una unidad de procesamiento de imagen para la generación de datos de imagen y/o movimientos de un objeto simulando una criatura viviente por la definición del objeto en un espacio virtual tridimensional, poniendo puntos blancos sucesivos en un espacio virtual tridimensional con respecto al objeto, calculando movimientos del objeto en correspondencia con cada punto blanco puesto sucesivo, generando los datos de imagen basados en el resultado calculado (Hiroshi, 2007).

La patente US00D687731S registrada en Estados Unidos con fecha 13 de agosto del 2013 por el inventor Peter David Dalton, reclama el diseño de un contador de peces ornamentales el cual lo describe con imágenes en diferentes perspectivas (David, 2013).

La patente JP20140208835 20141010 registrada con fecha 16 de mayo de 2016 por el inventor Akitomi et. al., se refiere a un contador de peces, un método de conteo de peces, un dispositivo de predicción para el número de peces, un método de predicción para el número de peces, un sistema de conteo de peces y un sistema de predicción para el número de peces (Akitomi Minoru, 2016).

La patente D532860 registrada en Estados Unidos con fecha 28 de noviembre de 2006 por los inventores Richardson y Sampson, reclama el diseño de un contenedor combinado de almacenamiento y contador de peces ornamentales, el cual lo describen en una imagen (Richardson, 2006).

La patente D310046 registrada en Estados Unidos con fecha 21 de agosto de 1990 por el inventor Matcham, reclama el diseño de un contador de peces ornamentales el cual lo describe con una imagen (Matcham, 1990).

### I.5.3 Contadores de peces comerciales

La figura 1 presenta un típico contador manual, es compacto y portátil, diseño elegante y resulta muy cómodo al momento de usarlo, la velocidad de la pulsación depende de la reacción humana, el diseño del contador es regularmente en forma cilíndrica con un diámetro de 45 mm, cuenta con un botón para el conteo sucesivo y con otro botón para el reinicio del conteo.



Figura 1: Contador manual.

Celeromics es un fabricante líder de sistemas de conteo de células para microscopios, sus productos permiten el conteo de células sin material adicional o reactivos, con bajo mantenimiento y actividad prolongada. Algunas de sus características son:

- Contador para uso en microscopio.
- Reduce la fatiga visual y la pérdida de visión.
- Elimina el error humano.
- Cuenta células y microorganismos de 1 a 1000 micras de diámetro.

- Guarda el historial del conteo y cultivos.
- Tiempo de procesamiento de 10 segundos.

La figura 2 presenta el contador de células para microscopios de la marca comercial Celeromics.

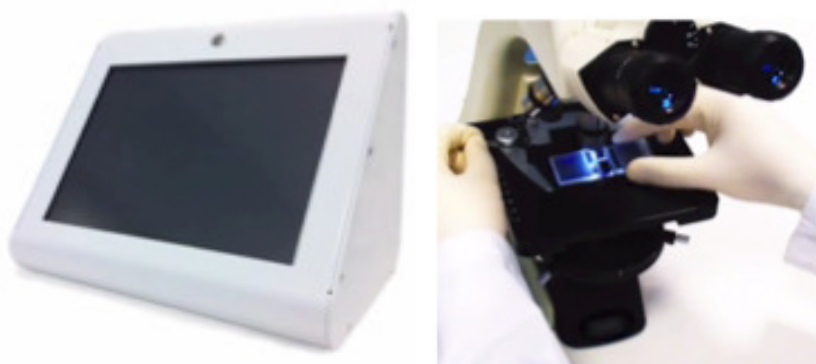


Figura 2: Contador de células para microscopios de la marca comercial Celeromics.

La empresa Faivre se formó en 1958 bajo la creación de un calibrador automático de truchas y lo llamaron “la Calibromatic”, desde entonces se han dedicado a realizar contadores para diferentes especies, tales como Truchas, Salmones y Tilapias. Actualmente comercializan tres tipos de contadores de peces: PescaVision 10 teniendo una capacidad máxima de 1.5 Toneladas/hora, PescaVision 30 con una capacidad máxima de 3 Toneladas/hora y PescaVision 50 con una capacidad máxima de 4 Toneladas/hora. La figura 3 presenta el contador Favre de especies marinas.

Los contadores de la empresa Vaky se utilizan con peces tan pequeños como los alevines hasta tamaños de peces adultos, por lo que cuenta con un amplio rango de tamaños de peces para su conteo. Cuenta principalmente con dos tipos de contadores llamados Macro Counter y Micro Counter. El Macro Counter tiene una capacidad de 200,000 salmones o 1 millón de alevines y tiene un área de conteo de 100 cm



Figura 3: Contador Faivre de especies marinas.

de ancho, el Macro Counter es ideal para la entrega precisa y rápida de salmones y alevines. El Macro Counter puede funcionar con un canal o con 4. El Micro Counter es excepcionalmente bueno para contar alevines. El área de conteo es de 50 cm de ancho con una capacidad de 500,000 gramos de pescado por hora. El Micro Counter es ideal para el manejo interno de pescado más pequeño o la entrega de alevines. El Micro Counter puede funcionar con un canal o con 3. Las principales ventajas de los contadores Vaki son: precisión, velocidad, verificación y peces en el agua en todo momento. La figura 4 muestra los contadores Vaki Macro Counter y Micro Counter.

## I.6 Organización del trabajo de tesis

Este trabajo de tesis de doctorado se organiza en cinco capítulos. El **segundo capítulo** presenta el marco teórico referente al fundamento matemático, los momentos invariantes de Hu, los fundamentos del procesamiento digital de imágenes, los sistemas embebidos y las especies marinas a contabilizar. El **tercer capítulo** presenta el sistema propuesto, la construcción de las estructuras mecánicas, la calibración del sistema, el algoritmo propuesto, la descripción del código Matlab/GNU Octave y la integración del sistema.



Figura 4: Contadores Vaki Micro Counter y Macro Counter.

El **cuarto capítulo** presenta los resultados empleando los momentos invariantes de  $H_u$ , los resultados empleando una PC con Matlab, los resultados empleando Raspberry Pi con GNU Octave y la comparación entre ambas plataformas. Finalmente, el **quinto capítulo** presenta las conclusiones generales, aportaciones y potenciales trabajos a desarrollar a futuro.

# Capítulo II

## Marco Teórico

### II.1 Introducción

El campo del tratamiento digital de imágenes ha aumentado considerablemente su estudio e investigación debido a temas como la morfología de las imágenes, las redes neuronales, el procesamiento de imágenes a color, la compresión, el reconocimiento de objetos y los sistemas inteligentes de análisis de imágenes. El interés por los métodos de tratamiento digital de imágenes deriva de dos áreas principales de aplicación: la mejora de la información pictórica para la interpretación humana y el procesamiento de los datos de la escena para la percepción autónoma por una máquina (Gonzalez y Woods, 2016). Es importante hacer notar que la descripción de un objeto puede basarse en sus bordes o contornos o bien, puede describirse con base en la forma de la región, o incluso a la distribución de la textura dentro de dicha región. Según (Gonzalez y Woods, 2016) hay dos formas de abordar la representación de una región: i) hacerlo en términos de sus características externas (básicamente su contorno) y ii) en términos de sus características internas, esto puede ser a través de los niveles de grises que comprenden la región. En este sentido el desarrollo de este capítulo se basa en el fundamento matemático necesario para mejorar la interpretación de las imágenes utilizadas durante el desarrollo de este trabajo de tesis, así como una breve descripción de los tipos de peces que se estarán analizando en las imágenes.

## II.2 Fundamentos de procesamiento digital de imágenes

Una imagen es representada en escala de grises por medio de una matriz bidimensional de  $m \times n$  elementos, en donde  $n$  representa el número de píxeles de ancho y  $m$  el número de píxeles de largo. El elemento  $v_{11}$  corresponde al elemento de la esquina superior izquierda la cual se puede ver en la figura 5, donde cada elemento de la matriz de la imagen tiene un valor de 0 (negro) a 255 (blanco) (Castillo, 2012).



Figura 5: Representación de una imagen en escala de grises.

En ocasiones es necesario hacer cálculos que requieren procesar por completo la imagen, en estos casos hacerlo sobre la resolución original de la imagen sería muy costoso (Moore, 2007) y (Pérez López, 2002). Una alternativa más eficiente, resulta el sub-muestreo de la imagen. Sub-muestreo significa generar una imagen a partir de tomar muestras periódicas de la imagen original, de tal forma que esta quede más pequeña. Si se considera la imagen  $I(m,n)$  definida como:

$$I(m, n) = \begin{bmatrix} v_{11} & v_{12} & v_{13} & \dots & v_{1n} \\ v_{21} & v_{22} & v_{23} & \dots & v_{2n} \\ v_{31} & v_{32} & v_{33} & \dots & v_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & v_{m3} & \dots & v_{mn} \end{bmatrix}$$

y se desea sub-muestrear la imagen para obtener la mitad de su tamaño original, la nueva imagen quedaría compuesta por los elementos tomando uno sí y otro no de la imagen original:

$$I_{s_2}(m, n) = \begin{bmatrix} v_{11} & v_{13} & v_{15} & \dots & v_{1(n-2)} \\ v_{31} & v_{33} & v_{35} & \dots & v_{3(n-2)} \\ v_{51} & v_{53} & v_{55} & \dots & v_{5(n-2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{(m-2)^1} & v_{(m-2)^3} & v_{(m-2)^5} & \dots & v_{(m-2)(n-2)} \end{bmatrix}$$

Por otro lado una imagen a color RGB, es representada por una matriz tridimensional  $m \times n \times p$ , donde  $m$  y  $n$  tienen el mismo significado que para el caso de las imágenes de escala de grises, mientras  $p$  representa el plano RGB, el cual puede ser 1 para el rojo, 2 para el verde y 3 para el azul. La figura 6 muestra detalles de estos conceptos.



Figura 6: Representación de una imagen a color en RGB.

Las matrices  $I_R$ ,  $I_G$  e  $I_B$  representan la imagen a color en RGB de forma matemática:

$$I_R(m, n, 1) = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{bmatrix} \quad I_G(m, n, 2) = \begin{bmatrix} g_{11} & g_{12} & \dots & g_{1n} \\ g_{21} & g_{22} & \dots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{m1} & g_{m2} & \dots & g_{mn} \end{bmatrix}$$

$$I_B(m, n, 3) = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \dots & b_{mn} \end{bmatrix}$$

## II.2.1 Histograma de imágenes

El histograma de una imagen digital con niveles de intensidad de gris en el rango  $[0, L - 1]$  es una función discreta  $h(r_k) = n_k$ , donde  $r_k$  es el  $k$ -ésimo nivel de intensidad de gris,  $n_k$  es el número de píxeles con intensidad  $r_k$  en la imagen y  $k = 0, 1, 2, \dots, L - 1$ , (Gonzalez y Woods, 2016).

De forma general se puede decir que  $p(r_k)$  da una idea del valor de la probabilidad de que aparezca el nivel de gris  $r_k$  en una imagen digital. La representación gráfica de esta función para todos los valores de  $k$  proporciona una descripción global de la apariencia de la imagen. Por ejemplo la figura 7 muestra el histograma de la imagen 5.

Los histogramas son la base de numerosas técnicas de procesamiento de imágenes. Su manipulación puede ser usada para mejorar la calidad de las imágenes, además proporcionan estadística útil de la imagen y es empleada para compresión y segmentación de imágenes digitales. Los histogramas son sencillos de calcular en software y también son factibles de implementar en hardware económico, por lo que son una herramienta popular para el procesamiento de imágenes en tiempo real (Gonzalez y Woods, 2016).

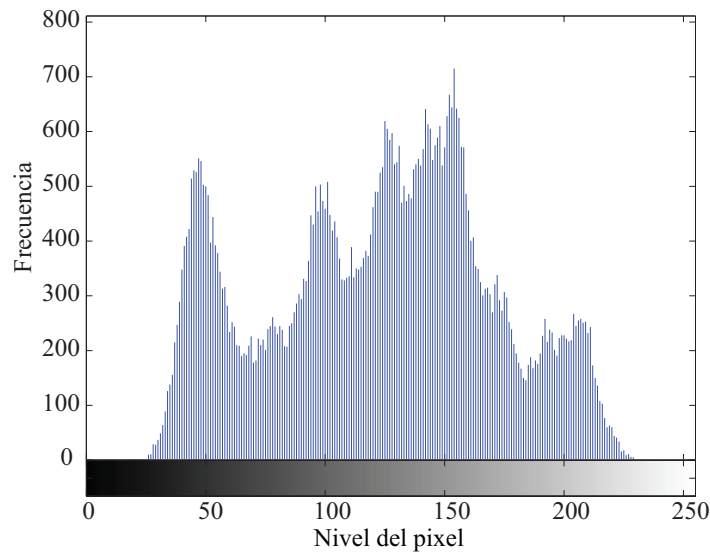


Figura 7: Histograma de la imagen digital mostrada en la figura 5.

## II.3 Fundamento matemático

Los pixeles de una imagen digital se pueden representar por una matriz  $A = [a_{ij}]_{m \times n}$  donde  $m$  y  $n$  son las filas y columnas respectivamente. El total de pixeles de la matriz  $A$  se puede estimar por la ecuación (1):

$$N = m \times n. \quad (1)$$

Los pixeles de la imagen digital representada por la matriz  $A$  deben estar dentro del rango de los niveles de grises  $[0, 1, 2, \dots, L - 1]$ , el número de pixeles en el nivel  $i$  se denotan por  $n_i$  y el número total de pixeles por  $N = n_1 + n_2 + n_3 + \dots + n_L$ . Según (Huang y Wang, 2009) y (Fan y Liu, 2013) con el fin de simplificar el análisis, se normalizan los niveles de grises del histograma considerándolos como una distribución de probabilidad (2):

$$p_i = \frac{n_i}{N}, \quad p_i \geq 0, \quad \sum_{i=1}^L p_i = 1. \quad (2)$$

Ahora, suponiendo que separamos los pixeles en dos clases  $C_0$  y  $C_1$  (fondo y objetos, o viceversa) mediante el umbral del nivel  $k$ ;  $C_0$  determina los pixeles con niveles  $[1, 2, \dots, k]$ , y  $C_1$  determina los pixeles con niveles  $[k + 1, \dots, L]$ . Entonces las probabilidades de ocurrencias de clase y los niveles medios de clase, respectivamente, están dados por (3) y (4):

$$\omega_0 = Pr(C_0) = \sum_{i=1}^k p_i = \omega(k) \quad (3)$$

$$\omega_1 = Pr(C_1) = \sum_{i=k+1}^L p_i = 1 - \omega(k) \quad (4)$$

así como (5) y (6)

$$\mu_0 = \sum_{i=1}^k iPr(i|C_0) = \sum_{i=1}^k \frac{iP_i}{\omega_0} = \frac{\mu(k)}{\omega(k)} \quad (5)$$

$$\mu_1 = \sum_{i=k+1}^L iPr(i|C_1) = \sum_{i=k+1}^L \frac{iP_i}{\omega_1} = \frac{\mu_T - \mu(k)}{1 - \omega(k)}, \quad (6)$$

donde

$$\omega(k) = \sum_{i=1}^k P_i \quad (7)$$

$$\mu(k) = \sum_{i=1}^k iP_i \quad (8)$$

son los momentos ascendentes acumulados del histograma desde el orden cero, el de primer orden y hasta el k-ésimo nivel respectivamente, y (9)

$$\mu_T = \mu(L) = \sum_{i=1}^L iP_i \quad (9)$$

es el nivel medio total de la imagen original.

Es importante señalar que el principio de clasificación de éste método es lineal, es decir, para realizar la tarea de discriminación se basa en el cálculo de un umbral calculado a partir de los histogramas de los niveles de grises de la escena de la imagen. Tal como se observa en las ecuaciones 1 al 9, las cuales nos permiten detectar a los peces dentro del acuario, esto se realiza mediante el principio de funcionamiento que el acuario contiene dos clases ( $\omega_0$  y  $\omega_1$ ), es decir, una clase son los peces y la otra clase es el fondo del acuario (background), se propone el uso de este método debido a que emplea operaciones matemáticas básicas (sumas, restas, multiplicaciones y divisiones), con lo que se logra una gran ventaja por ser ligero desde el punto de vista computacional para ser implementado y ejecutado por el sistema embebido propuesto, como una

primera fase para validar experimentalmente su viabilidad. Los resultados de exactitud reportados en (Huang y Wang, 2009) y (Fan y Liu, 2013) son similares a los obtenidos con el método propuesto en este trabajo de tesis. Por otra parte, con respecto a las características utilizadas para la clasificación se propone que sea el área y el perímetro de los peces tal como se especifica en el algoritmo propuesto que se muestra en la figura 25.

## II.4 Momentos invariantes de Hu

En (Fix y Hodges Jr, 1951) y (Fix y Hodges Jr, 1952) se utilizan por primera vez métodos de identificación de patrones basados principalmente en métodos estadísticos, los cuales crean las bases para lo que posteriormente se conoce como reconocimiento de patrones. Uno de los pioneros en plantearse la aplicación de momentos estadísticos y que a la vez estuvieran normalizados a la rotación, traslación y escala es (Hu, 1962), quien propone un conjunto de siete momentos invariantes. Éstos momentos se encuentran a su vez, basados en los momentos centrales normalizados, reformulados por Hu para permitir la invariancia rotacional. Para esta tarea se emplean dos métodos: ejes principales y momentos invariantes absolutos, derivando expresiones de invariantes algebraicas aplicadas a la generación de momentos bajo transformaciones rotacionales. Hu define los momentos normalizados no escalados como:

$$v_{pq} = \frac{\mu_{pq}}{(\mu_{00})^\gamma}. \quad (10)$$

Siendo

$$\gamma = 1 + \frac{p+q}{2}. \quad (11)$$

Donde  $p$  y  $q$  son las dimensiones de la imagen, dando lugar al conjunto de siete momentos los cuales son invariantes a la traslación, rotación y escala:

$$\phi_1 = \mu_{20} + \mu_{02}. \quad (12)$$

$$\phi_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2. \quad (13)$$

$$\phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2. \quad (14)$$

$$\phi_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2. \quad (15)$$

$$\begin{aligned} \phi_5 = & (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ & + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2]. \end{aligned} \quad (16)$$

$$\begin{aligned} \phi_6 = & (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\ & + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}). \end{aligned} \quad (17)$$

$$\begin{aligned} \phi_7 = & (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ & - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2]. \end{aligned} \quad (18)$$

Donde

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - x_c)^p (y - y_c)^q f(x, y) dx dy. \quad (19)$$

es el momento central del objeto  $f(x, y)$  y  $(x_c, y_c)$  son las coordenadas del centroide.

## II.5 Sistema embebido Raspberry Pi

El avance tecnológico actual desarrolla computadoras de placa reducida (Single Board Computer), que son computadoras completamente integradas y de alto rendimiento en un solo circuito (System On a Chip). Tal es el caso de la denominada Raspberry Pi, que desde sus inicios en el 2006 y hasta su modelo más reciente lanzado en 2016, (Raspberry Pi 3) ha tenido una muy buena aceptación (Ibrahim, 2014).

La Raspberry Pi o Raspi como se conoce mundialmente, es la tendencia hoy en día, esto debido a sus características de diseño. Básicamente la Raspi es una computadora del tamaño de una tarjeta de crédito, cuenta con una memoria RAM, un CPU Quad Core marca ARM, red Ethernet por cable y puertos USB. La Raspi fue desarrollada como un equipo económico con el fin de estimular en los jóvenes el gusto por la programación y computación (Watson *et al.*, 2014).

En este trabajo de tesis, para realizar el conteo se utiliza una Raspberry Pi 2 Modelo B, la cual basa su diseño en un SoC Broadcom BCM2836, que incluye un procesador Quad Core ARM Cortex A7 de 900 MHz, un GPU VideoCore IV y una memoria SDRAM de 1 GB. La Raspi ejecuta sistemas operativos basados en el Kernel de Linux. Debido a que tiene un procesador ARM Cortex A7 puede ejecutar toda la gama de GNU/Linux incluyendo Snappy Ubuntu Core. Las dimensiones de la Raspi son: 14.22 x 9.9 x 4 cm. La figura 8 muestra una Raspberry Pi 2 modelo B empleada para el desarrollo de este trabajo de tesis.

Dado que la Raspi no cuenta con un disco duro integrado, se utiliza una tarjeta de memoria Ultra microSDHC UHS-I SanDisk clase 10 con una capacidad de almacenamiento de 32 GB y una velocidad de lectura de hasta 48 MB/s para la transferencia rápida de archivos. La figura 9 muestra la memoria clase 10 empleada en este trabajo de investigación.

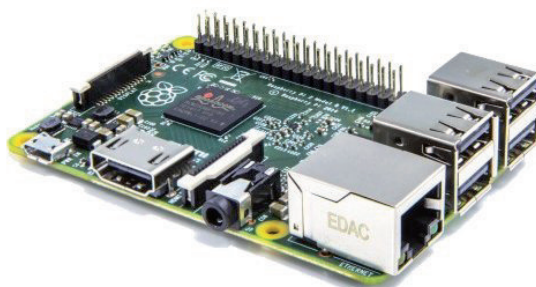


Figura 8: Raspberry Pi 2 Modelo B.



Figura 9: Memoria microSD clase 10 de 32 GB.

Esta tarjeta de memoria es resistente al agua, a los rayos X, al magnetismo y a los golpes. Las dimensiones de la tarjeta de memoria son: 0.76 x 1 x 1.5 cm.

La cámara de la Raspberry Pi tiene una resolución de 5 megapíxeles, fabricada con tecnología CMOS y lente de foco fijo. Su resolución es de 2592 x 1944 píxeles en imágenes estáticas y en video es capaz de soportar 1080p30, 720p60 y 640x480p60/90. Se acopla a la Raspberry Pi utilizando la interfaz dedicada CSI diseñada especialmente para la conexión de cámaras. Sus dimensiones son de 25 x 20 x 9 mm, y su peso es de 3 g. La figura 10 muestra la cámara de la Raspberry Pi 2 Modelo B de 5 MP.

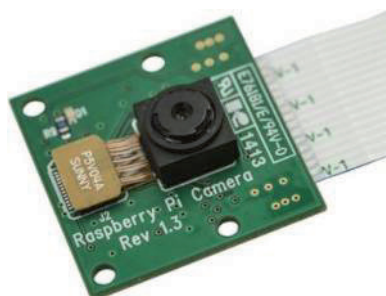


Figura 10: Cámara para Raspberry Pi 2 Modelo B de 5 MP.

## II.6 Especies marinas a contabilizar

Para la realización de este trabajo de tesis se utilizaron 2 diferentes especies de peces: Guppies (*Poecilia reticulata*) y Mollies (*Poecilia sphenops*) de las cuales se presenta una breve descripción.

### II.6.1 Guppies (*Poecilia reticulata*)

El guppy, lebistes o pez millón (*Poecilia reticulata*), es un pez ovovivíparo de agua dulce procedente de Sudamérica que habita en zonas de corrientes bajas de ríos, lagos y charcas. Es originario de Trinidad, Barbados, Venezuela y norte de Brasil. Los guppies son peces de agua caliente y se recomienda criarlos en temperaturas entre los 22°C y 28°C (óptima: 25°C), aunque son peces muy tolerantes con las condiciones del agua, siendo posible criarlos a temperatura ambiente si ésta no se aleja mucho de este intervalo.

Por otra parte, en las peceras es recomendable que se usen filtros y oxigenadores para estabilizar el agua, ya que esto ayuda a los Guppies a que su promedio de vida suba considerablemente.

Los machos adultos miden alrededor de 2 cm, si bien pueden llegar a medir hasta 4 cm en algunos casos. Las hembras son siempre más grandes, miden alrededor de 3 cm o incluso pueden llegar a medir 5 cm máximo en las variedades más grandes. Los Guppies machos se destacan frente a otros peces de acuario de mayor tamaño por sus bellos colores verdes, azules, rojos e incluso atigrados. Las hembras son generalmente de color aceituna y solo suelen mostrar color en su aleta dorsal y cola (Villarejo, 2002).

Esta especie tiene una esperanza de vida de aproximadamente tres años en condición estándar, que compensan con unas capacidades reproductivas excepcionales. En la figura 11 se muestran diferentes peces Guppies.



Figura 11: Peces Guppies (*Poecilia reticulata*).

## II.6.2 Mollies (*Poecilia sphenops*)

Los Mollies se encuentran desde México hasta Colombia, son una especie de agua dulce, acepta temperaturas entre los 19°C y los 30°C, siendo lo ideal entre 22°C y 28°C. Debido a su tamaño, estos peces son aptos para acuarios de diversos tamaños, desde pequeños a grandes, siempre que tengan espacio suficiente para nadar con libertad. Los Mollies machos pueden alcanzar los 6 cm de longitud total y las hembras los 10 cm. Las hembras son de mayor tamaño que los machos, mientras que estos son más alargados y tienen las aletas más desarrolladas. Es omnívoro, puede alimentarse de cualquier tipo de alimento: vivo, disecado, escamas, etc. Para esta especie es necesaria una fuente de alimento de origen vegetal. En el caso de los alevines, la alimentación consiste en pequeñas pulgas de agua, aunque también aceptan escamas trituradas (Moyle y Joseph, 2004).

Su comportamiento en general es pacífico, sin embargo los machos adultos pueden volverse territoriales con otros peces del acuario. Esta especie realiza una actividad de limpieza de las algas del acuario. Es una especie apropiada para un acuario comunitario en el que conviven otras especies de tamaño similar (Nelson *et al.*, 2016). La figura 12 muestra una vista lateral de un pez molly hembra.



Figura 12: Pez Molly hembra (*Poecilia sphenops*).

# Capítulo III

## Desarrollo del sistema propuesto

El primer sistema propuesto en este trabajo de tesis doctoral consta de una Computadora Personal (PC), una cámara digital de 10.1 Megapíxeles, un teclado, un monitor, un ratón y un acuario donde están ubicados los peces a contabilizar, la figura 13 muestra el diagrama a bloques del primer sistema propuesto.

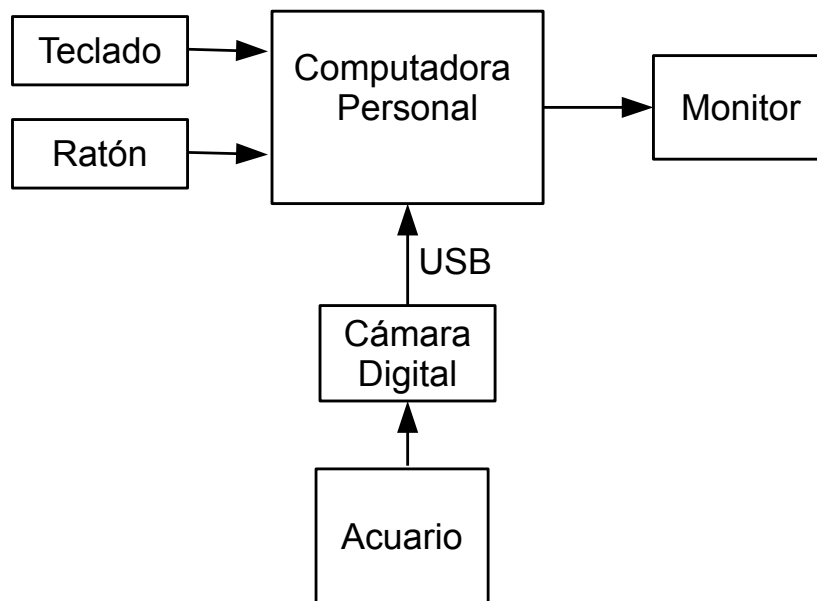


Figura 13: Diagrama a bloques del primer sistema propuesto.

### III.1 Construcción de las estructuras metálicas

De las necesidades derivadas de la implementación física del dispositivo contador de peces, en la figura 14 se muestra la primer estructura metálica diseñada y construida. La primer estructura construida está formada por un marco de metal que soporta una

cámara digital Sony Cyber-shot DSC-W180 con resolución de 10.1 Megapíxeles con conexión USB y un sistema de iluminación que consiste de dos lámparas fluorescentes de policarbonato FLC-16W/65 de 16 W @ 110  $V_{ac}$  cada una y cuya iluminación se distribuye uniformemente sobre la superficie de un contenedor de vidrio (acuario) a través de un difusor de acrílico. Las dimensiones de la estructura metálica como se indican en la figura 14 son: 60 x 20 x 40-50 cm de largo, ancho y alto. La variación en altura permite encontrar el mejor foco para la captura de imágenes. La parte superior de la estructura metálica contiene una base para colocar la cámara digital. El acuario es un contenedor de vidrio rectangular de: 8 x 42 x 33 cm. de alto, largo y ancho respectivamente. Es en este contenedor donde se alojan o depositan los peces vivos para su conteo; las pruebas experimentales sugieren un nivel de agua estrecho de 3 cm. de altura aproximada, que equivalen a 4.158 litros, la figura 14 muestra el acuario desarrollado.

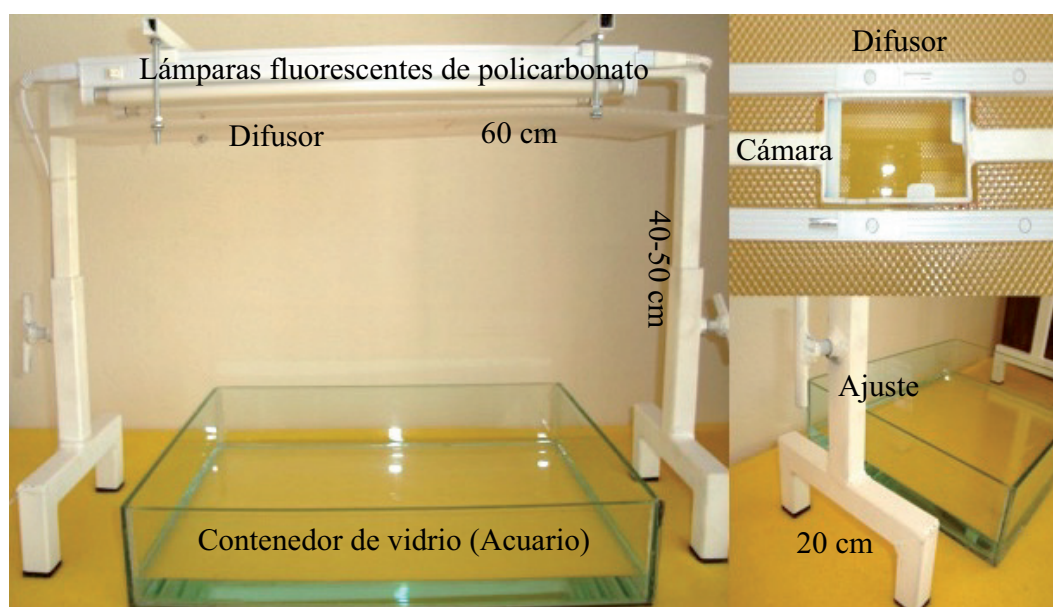


Figura 14: Primera estructura metálica del contador de peces propuesto.

El segundo sistema propuesto en este trabajo de tesis doctoral consta de una Raspberry Pi 2, una cámara CCD de 5 MP, un teclado, un ratón, un monitor HDMI y un acuario donde están ubicados los peces a contabilizar, la figura 15 muestra el diagrama a bloques del segundo sistema propuesto.

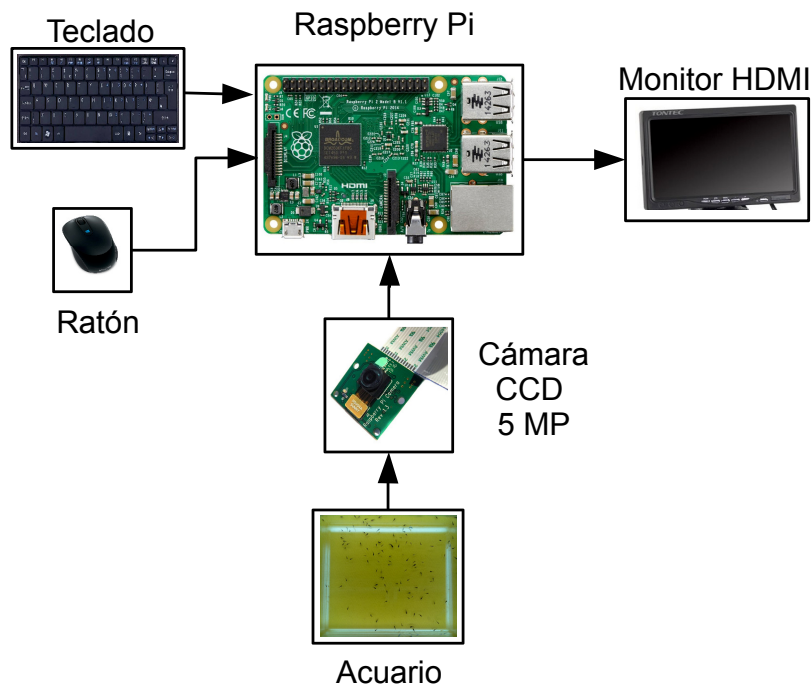


Figura 15: Diagrama a bloques del segundo sistema propuesto.

La segunda estructura metálica construida es la que se muestra en la figura 16. La estructura está formada por un marco de metal que soporta la Raspberry Pi, la cámara digital y un sistema de iluminación que consiste de cuatro lámparas fluorescentes de policarbonato, dos de ellas son FLC-16W/65 de 16 W @ 110  $V_{ac}$  y las otras dos son FCL-4006 de 8W @ 110  $V_{ac}$ . La iluminación se distribuye uniformemente sobre la superficie de un contenedor de vidrio (acuuario). Las dimensiones de la estructura metálica como se indican en la figura 16 son: 60 x 20 x 49-55 cm. de largo, ancho y alto. La variación en altura permite encontrar el mejor foco para la captura de imágenes. El **Apéndice A** presenta mayores detalles de la estructura mecánica.

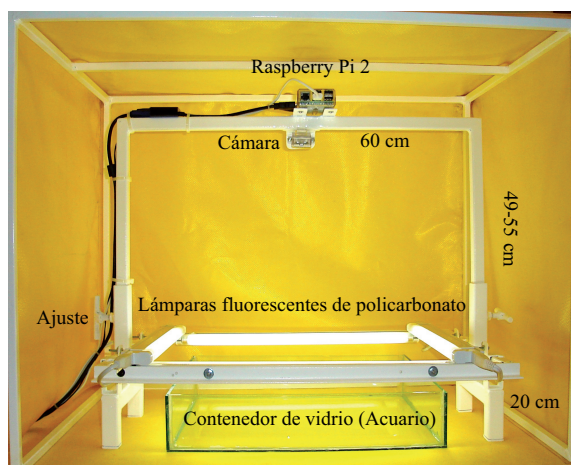


Figura 16: Segunda estructura metálica del contador de peces propuesto.

El sistema cuenta con una carcasa de lona de color amarillo con las siguientes dimensiones: 80 x 50 x 60 cm de largo, ancho y alto respectivamente. La carcasa de lona sirve para controlar parcialmente la iluminación al interior del sistema dado que cubre la parte superior del sistema así como la parte trasera y los laterales, permitiendo al usuario observar el contenedor de vidrio solo por la parte frontal. En la figura 17 se puede observar la carcasa de lona de color amarillo, mismo que se obtiene a partir de un estudio empírico buscando la optimización del color de fondo, el cual se presenta en la sección III.2.

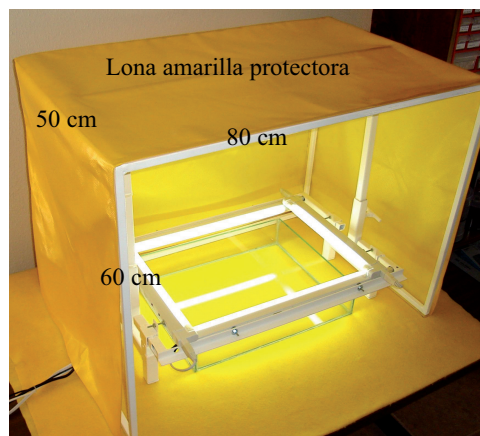


Figura 17: Vista superior de la segunda estructura con la carcasa de lona amarilla protectora del exceso de luz exterior.

El acuario es un contenedor de vidrio rectangular de: 8 x 42 x 33 cm de alto, largo y ancho respectivamente, este acuario tiene una capacidad de 11.088 litros. Es en este contenedor donde se alojan o depositan los peces vivos para su conteo, las pruebas experimentales sugieren un nivel de agua estrecho mínimo de 3 cm de altura aproximada. La figura 18 muestra el acuario construido para el contador de peces.

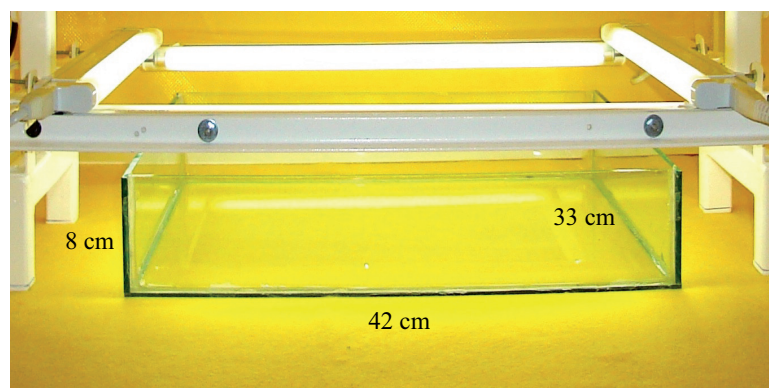


Figura 18: Acuario desarrollado para el sistema contador de peces.

La figura 19 muestra una toma superior donde se aprecia la Raspberry Pi montada sobre la estructura metálica orientada hacia el acuario que se encuentra en la parte inferior de la cámara digital.

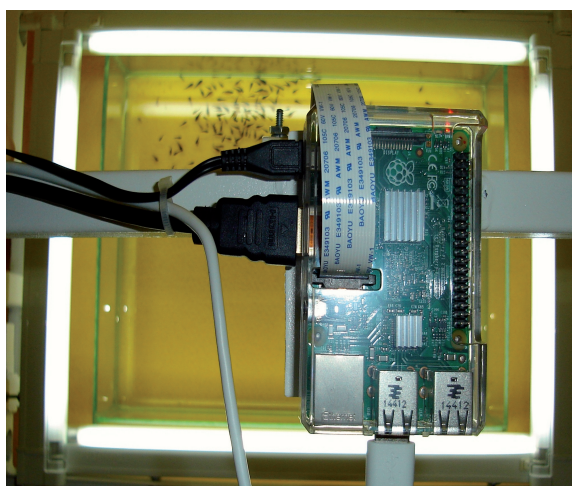


Figura 19: Raspberry Pi montada en la estructura metálica.

## III.2 Adquisición de imágenes digitales

La etapa que soporta al sistema contador de peces propuesto, es el proceso de adquisición de imágenes. Son muchas las variables involucradas para lograr capturar imágenes de buena calidad, que contribuyan al procesamiento computacional óptimo a fin de obtener la cantidad exacta de peces presentes en tiempo real, a partir de la captura de una sola imagen. Entre las variables involucradas se encuentran: tamaño de los especímenes, tipo de especie, movilidad y comportamiento grupal, nivel de profundidad de agua en el acuario, área del acuario, grado y tipo de iluminación, nivel de resolución de la cámara fotográfica, cantidad de luz, etc. Bajo la correlación de estas variables, se busca lograr el máximo contraste para cada una de las imágenes obtenidas. Es en esta dirección que el color de fondo resulta ser crucial en combinación con el sistema de iluminación que se implementa y que debe iluminar uniformemente toda el área del acuario. Las lámparas se ubican para que la iluminación sea lo suficientemente uniforme en la zona de interés y que su nivel de radiación sea moderada al grado que no afecte el comportamiento natural de los peces. Por otra parte, para definir el color óptimo de fondo del sistema, resulta necesario hacer referencia al espectro visible, al cual ordinariamente se le refiere como luz (Frenzel, 2015). Se realizaron dos pruebas empíricas que permiten definir el color óptimo del fondo del acuario, de entre la gama de colores: amarillo, rosa, naranja, café, azul, blanco, morado y verde, teniendo tanto el amarillo como el azul en dos tonalidades distintas. Las pruebas teóricas se basan en:

i) la obtención del promedio de los pixeles definida por (20):

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n a_i \quad (20)$$

y ii) el uso de la distancia Euclidiana definida por (21) para obtener el grado de error sistemático que presentan los colores:

$$d_E(P, Q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (21)$$

Donde  $P$  y  $Q$  son los puntos a medir y  $p_i$  y  $q_i$  son sus coordenadas cartesianas respectivas.

Dado que durante el desarrollo de este trabajo de investigación se realizaron dos sistemas, en la tabla I se muestran los resultados de análisis numérico usando una PC con Matlab y en la tabla II se muestran los resultados usando Raspberry Pi 2 con GNU Octave, ambas tablas muestran en la primer columna los colores usados, en la segunda columna el valor promedio de pixeles de las imágenes de fondo y en la tercer columna el error sistemático.

Tabla I: Valor promedio de pixeles de las imágenes de fondo del acuario obtenidos con una PC usando Matlab.

<b>Colores de fondo</b>	<b>Valor promedio de pixeles de las imágenes de fondo</b>	<b>Error sistemático</b>
Rosa	129.0592	1.0592
Amarillo canario	132.7983	4.7983
Amarillo huevo	135.9039	7.9039
Verde	118.8586	9.1414
Naranja	139.4497	11.4497
Morado	109.3018	18.6982
Blanco	147.333	19.333
Azul imperial	99.9878	28.0122
Azul rey	86.168	41.832
Café	77.5695	50.4305

Tabla II: Valor promedio de pixeles de las imágenes de fondo del acuario obtenidos con Raspberry Pi 2 y GNU Octave.

Colores	Valor promedio de pixeles de las imágenes de fondo	Error sistemático
Rosa	118	10
Amarillo canario	115	13
Amarillo huevo	119	9
Verde	101	27
Naranja	118	10
Morado	105	23
Blanco	118	10
Azul imperial	105	23
Azul rey	242	114
Café	242	114

Tanto la tabla I, como la tabla II, muestran los valores promedio de pixeles de las diferentes imágenes de fondo utilizadas para encontrar el color óptimo para el conteo minimizando errores, estas imágenes son tomadas en condiciones de iluminación distribuida uniformemente y sin peces en el acuario. Mediante el uso de la ecuación (21), ambas tablas muestran el error sistemático presente en los diferentes colores, en la tabla I el error es menor en los colores rosa, amarillo canario y amarillo huevo, y en la tabla II el error es menor en los colores amarillo huevo, naranja, blanco y rosa.

Estos valores promedio de los pixeles de los diferentes colores de fondo se pueden observar en el eje vertical de la figura 20, la cual muestra una superposición entre una señal analógica y una digital (rango de pixeles digitales). Cuando la señal analógica se aproxima o cruza el nivel de saturación superior (color blanco 255) y/o inferior (color negro 0) genera mayor desviación (error sistemático) respecto al resultado del promedio de sus pixeles. La señal digital (de la imagen de fondo) se debe mantener centrada (promediada) en 128, lo cual indica que su error en cuanto al promedio de sus pixeles será mínimo (inherente al sistema) y no presentará saturación, ni pérdida de información. De acuerdo a las pruebas experimentales, fue el amarillo canario y huevo

quienes mejor resultados (más cercanos al 128) arrojaron. De las tablas I y II se infiere que el mejor color es el amarillo huevo debido a que propicia el menor error sistemático, no descartando la posibilidad como trabajo futuro de realizar los mismos experimentos con los otros 3 colores que también presentaron reducidos errores sistemáticos.

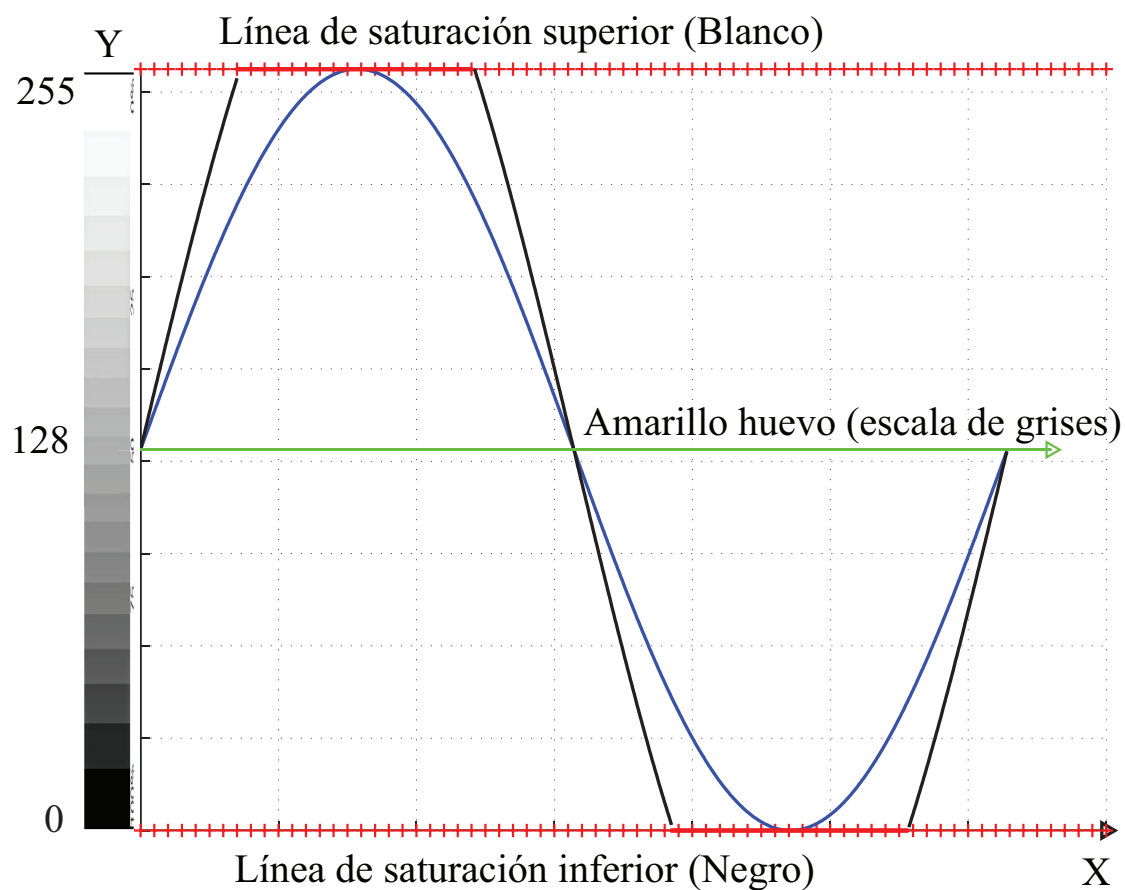


Figura 20: Analogía de señal con rango de valores de píxeles.

### III.2.1 Calibración del sistema propuesto

Para la puesta en operación de ambos sistemas contadores de peces, implicó que todos los elementos que los conforman trabajen en concordancia, lo que conlleva a la calibración propia de cada instrumento. Una parte fundamental lo constituye el enfoque de la cámara fotográfica (ya sea la Sony Cyber-shot o la Raspi Cam) sobre el área del acuario, que debe mantenerse sin variación a lo largo de un proceso de conteo. La línea de vista que debe mantener la cámara digital al capturar una imagen debe ser tal que abarque solamente el área de la base interna del acuario. En la figura 21 se muestra la imagen original capturada por la Raspberry Pi, en donde se aprecia el marco del acuario lo cual provoca un error sistemático y es no deseado, dicho error estaba presente en ambas cámaras digitales (usando PC con Matlab y Raspberry Pi con GNU Octave) siendo la solución recortar digitalmente la imagen, de forma tal que solo sea visible el área de la base interna del acuario, tal como se puede observar en la figura 22. Cabe mencionar que en la figura 21 se observa el reflejo de la luz blanca lo cual genera un pequeño error sistemático en el conteo final de peces.

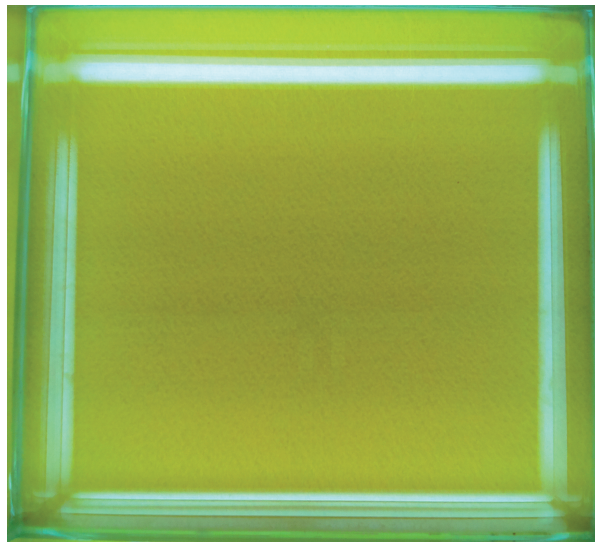


Figura 21: Enfoque no deseado de la imagen de entrada.

La figura 21 se recorta digitalmente con el fin de que no aparezca el marco del acuario y se logre que el sistema entregue resultados más confiables, aunque se aprecia que el reflejo de la luz blanca sigue presente, lo cual sigue generando un error mínimo en el conteo final de peces. El área requerida es cuando solo se obtiene el área interna del acuario como se ilustra en la figura 22.

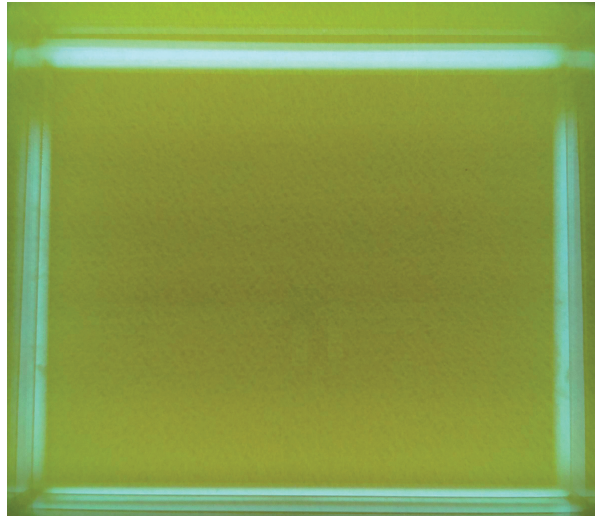


Figura 22: Enfoque calibrado de la imagen capturada en donde solo aparece la base del acuario.

### **III.3 Algoritmos propuestos**

#### **III.3.1 Identificación y conteo utilizando momentos invariantes de Hu (Primer algoritmo)**

Durante el desarrollo de este trabajo de tesis se implementan principalmente dos algoritmos para el conteo de peces, el primero se encuentra basado en los momentos invariantes de Hu (Hu, 1962), dicho algoritmo contabiliza e identifica a la vez, las especies de peces presentes en una imagen. La figura 23 muestra el diagrama de flujo del primer algoritmo.

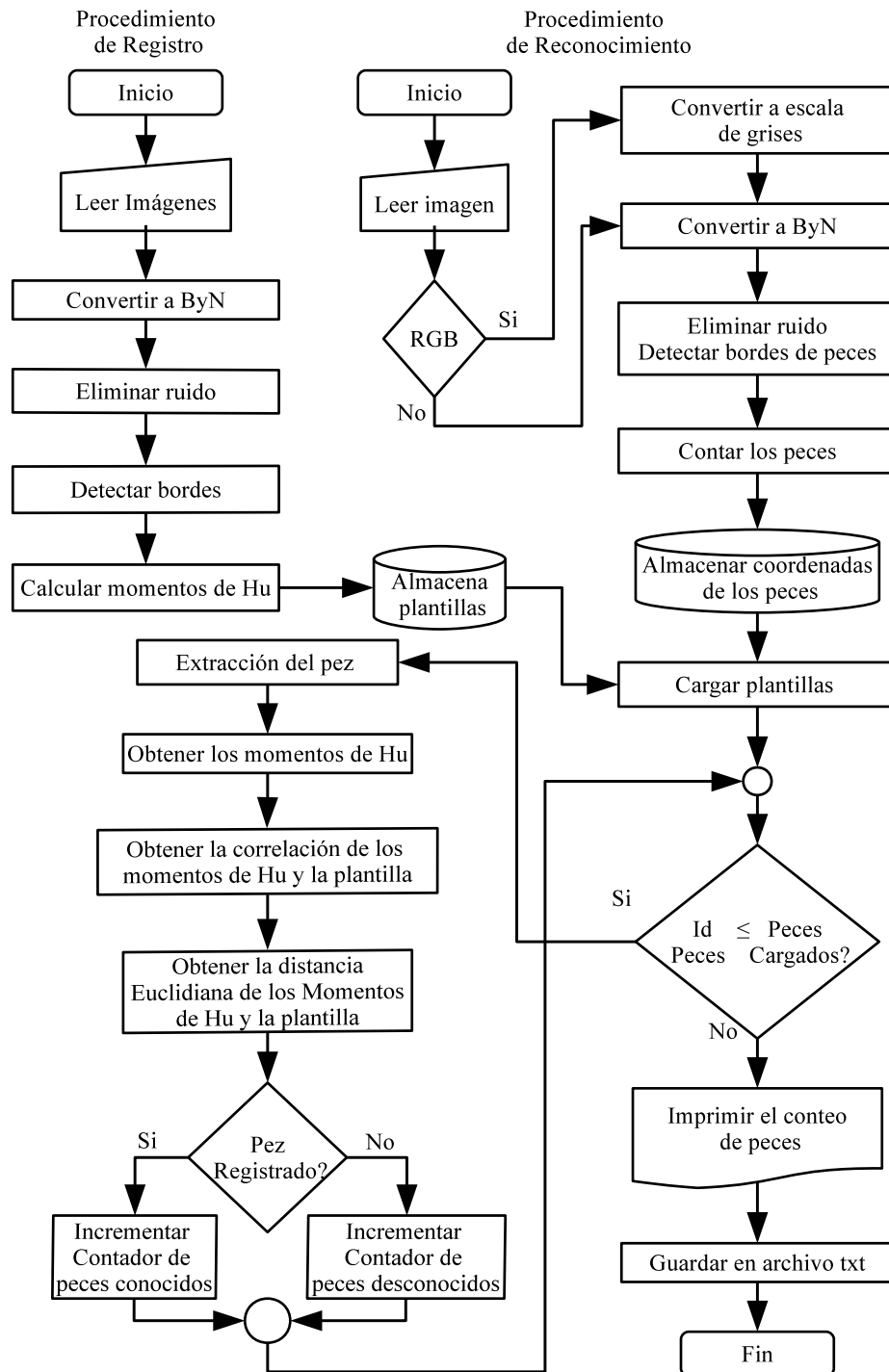


Figura 23: Diagrama de flujo del algoritmo basado en los momentos invariantes de Hu.

El programa recibe una imagen principal de entrada la cual puede ser a color (RGB) o en escala de grises, se reduce el ruido mediante el uso de la morfología en una imagen binaria usando una conectividad de 8 y se realiza un procesamiento para encontrar cuantos peces en este caso, existen dentro de ella (Gonzalez y Woods, 2016). Una vez que se obtienen los datos de la cantidad de peces, se extrae un pez a la vez y se calculan sus siete momentos de Hu (Hu, 1962), los cuales servirán para identificar a que especie corresponde el pez reconocido. El programa realiza la identificación del pez mediante dos procesos, el primero se basa en la correlación con una imagen extraída de una plantilla que cuenta con los momentos de las imágenes a identificar y que en éste caso corresponde a los peces que se muestran en la figura 24.

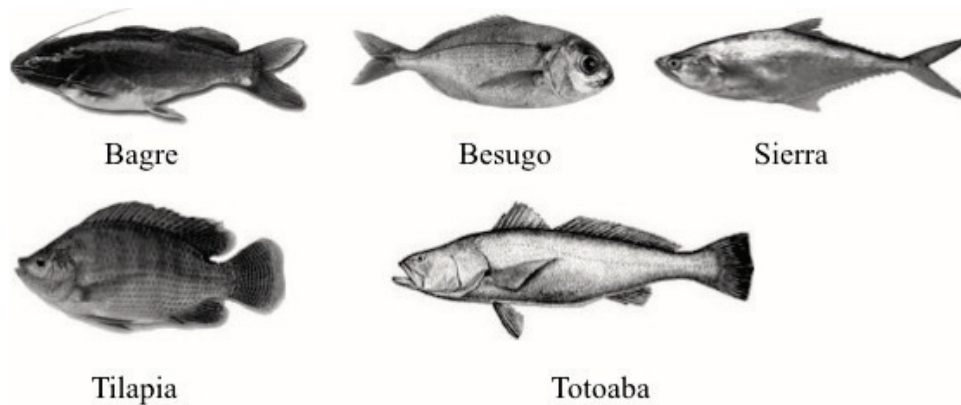


Figura 24: Plantilla base de peces a identificar.

La ecuación de la correlación utilizada es:

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}}. \quad (22)$$

Donde  $A$  y  $B$  son coeficientes a los que se les calcula la correlación, pudiendo ser ambos vectores o matrices siempre del mismo tamaño,  $\bar{A}$  y  $\bar{B}$  corresponden a los promedios aritméticos de  $A$  y  $B$  respectivamente (Kenney John, 1939).

El otro proceso de identificación o reconocimiento de la imagen implementada en el algoritmo propuesto es la distancia Euclidiana, formulada por la siguiente ecuación:

$$d_E(P, Q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}. \quad (23)$$

Donde  $P$  y  $Q$  son los puntos a medir y  $p_i$  y  $q_i$  son sus coordenadas cartesianas respectivas.

Particularmente en este primer algoritmo se propone que los peces a reconocer sean: Bagre, Besugo, Sierra, Tilapia y Totoaba y que forman la plantilla base de identificación presentada en la figura 24.

A partir de las ecuaciones 22 y 23, el programa entrega dos datos para la toma de decisión en cuanto a la identificación de un pez se refiere. Cuando en la imagen original se identifica un pez con su correspondiente en la plantilla base (figura 24), el valor de la correlación es muy cercano a 1, mientras que el valor de la distancia Euclidiana es muy cercano a 0. El factor de ponderación que toma en cuenta el algoritmo para definir a que especie corresponde el pez a identificar, es el valor de la menor desviación que entregue el cálculo para la correlación y la distancia Euclidiana.

### **III.3.2 Implementación del segundo algoritmo propuesto en Matlab/GNU Octave**

Dado que el algoritmo presentado en la sección III.3.1 solo hace identificación y conteo de peces basado en una imagen ideal, resulta necesario realizar algunas adecuaciones para contabilizar peces reales, por lo que el segundo algoritmo implementado en este trabajo se enfoca únicamente en contabilizar peces vivos depositados dentro de un acuario. Este segundo algoritmo, primero se implementa en la plataforma de Matlab y posteriormente se migra a Raspberry Pi empleando GNU Octave, por lo que en el capítulo IV se muestran los resultados del funcionamiento en ambas plataformas.

El segundo algoritmo propuesto también se encuentra basado en un procesamiento digital de imágenes, a fin de estimar la cantidad de peces vivos dentro del acuario. Esto se lleva a cabo a partir de procesar la imagen capturada en tiempo real de la distribución que guardan los peces bajo análisis. Al igual que en el primer algoritmo propuesto, la imagen de entrada al programa puede ser a color (RGB) o en escala de grises. Las imágenes capturadas a lo largo de todo un proceso de conteo son diferentes y cada una tienen condiciones particulares, mismas que el segundo algoritmo propuesto toma en consideración para dar certidumbre a la estimación de la cantidad de peces. En las imágenes capturadas se manifiestan alteraciones debido a varios factores: i) vibraciones mecánicas externas, ii) movimiento del agua del acuario, iii) movimiento propio de los peces, iv) niveles de sombra generados por los peces, v) grado de turbidez del agua del acuario, vi) reflexiones internas debidas a las paredes del acuario, etc. Para cada una de estas condiciones el algoritmo desarrollado realiza un proceso de filtrado de tipo binario y hace uso de la morfología sobre imágenes binarias aplicando una conectividad de 8 para determinar con ello si dos píxeles son vecinos y se encuentran conectados en direcciones perpendiculares (arriba, abajo, derecha, izquierda) y/o diagonalmente (Gonzalez y Woods, 2016).

La figura 25 muestra el diagrama de flujo del segundo algoritmo propuesto. La imagen capturada de la distribución de los peces, es la entrada digital al algoritmo, esta se acondiciona a una imagen en escala de grises para que el algoritmo la convierta posteriormente en blanco y negro y pueda realizar el filtrado de ruido, en todo momento se toma como base la conectividad 8 (Huang y Wang, 2009).

Una vez que la imagen se encuentra en blanco y negro se procede a encontrar los componentes de interés (peces o ruido), posterior a esto se obtienen las propiedades de dichos componentes, una de estas propiedades sirve para poder enmarcar en rectángulos verdes los peces y la otra propiedad sirve para obtener el área de cada componente.

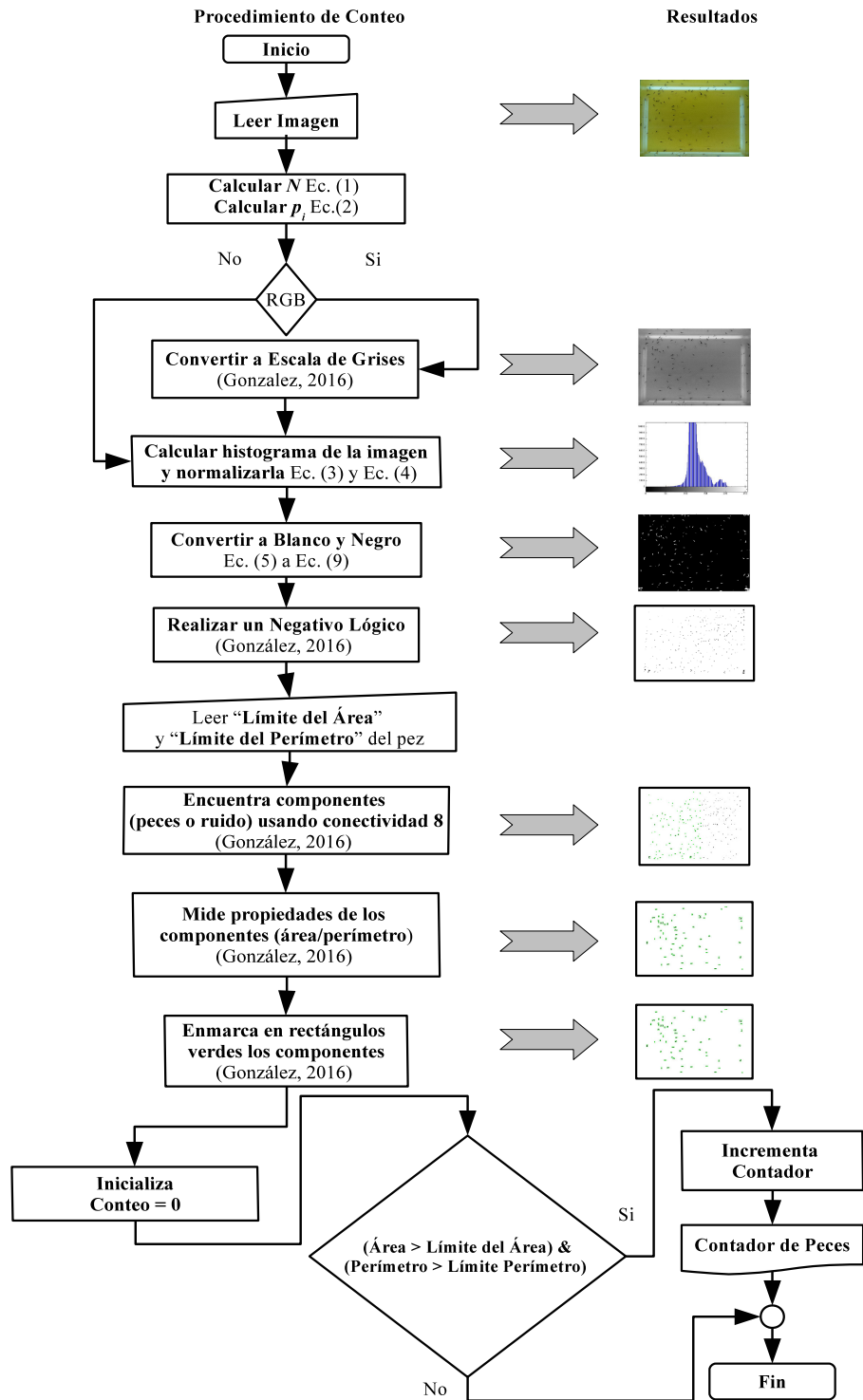


Figura 25: Diagrama de flujo del segundo algoritmo propuesto.

Una vez que se tienen identificados todos los componentes y con su área enmarcada, se inicia el contador de peces y se delimita el umbral (con base en el área de los peces) que sirve para contarlos, dicho umbral es variable dependiendo del tamaño de los peces. El algoritmo incrementa el contador cada que un pez es de tamaño mayor al umbral pre-establecido, finalmente se despliega en pantalla el resultado de dicho conteo.

Como prueba inicial de la operatividad y calibración del algoritmo desarrollado en concordancia con el sistema de adquisición de imágenes, se establece que al procesar una imagen sin peces presentes en el acuario, el resultado del conteo debe ser cero. Esta condición básica de operación del contador de peces se logra con un color de fondo amarillo en combinación con el sistema de iluminación y con la calibración descrita en la sección III.2.1.

En la figura 26 se muestra visualmente el resultado de procesar una imagen del acuario sin peces. La figura 26A corresponde a la imagen capturada por la cámara del fondo del acuario donde se aprecia predominantemente el color amarillo huevo y en la cual se puede observar que el marco de vidrio ya fue recortado digitalmente. Esta imagen al binarizarse por el algoritmo, genera una imagen carente de información (imagen negra) como se observa en 26B. Acorde con la secuencia lógica del procesamiento de la información de la imagen de entrada, se invierten los colores de la figura 26B obteniéndose una imagen completamente blanca, tal como se observa en la figura 26C. Finalmente en la etapa de conteo, el algoritmo al procesar una imagen en blanco (figura 26C) entrega en consecuencia un valor de cero peces presentes, confirmando que el sistema ha sido calibrado y el error sistemático es cero.

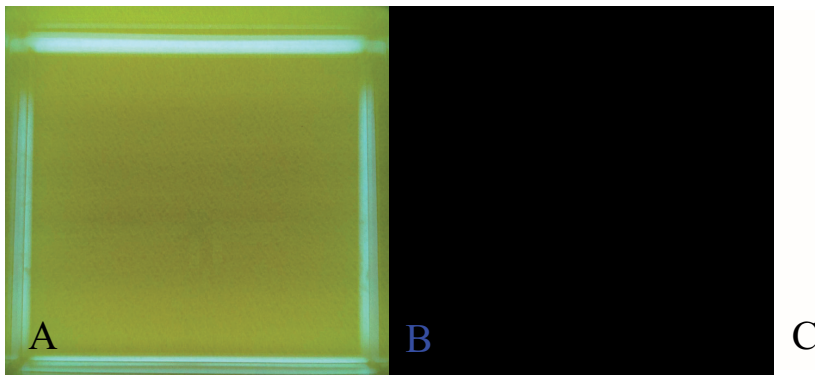


Figura 26: Visualización del procesamiento de una imagen del acuario sin peces, con fondo color Amarillo Huevo. A) Imagen de entrada. B) Imagen binarizada. C) Imagen complementada.

La concordancia entre la operatividad del algoritmo desarrollado y el sistema de iluminación del sistema contador de peces representado en la figura 26, cambia de manera radical con el cambio de color de fondo. En la figura 27, se presenta el procesamiento de la misma imagen manejada que en la figura 26 (acuuario sin peces) pero con color blanco de fondo (figura 27A). Al comparar las figuras 27B y 26B que corresponden a la etapa de binarizar la imagen de entrada, se observa el efecto que se tiene al utilizar un fondo de color blanco en el acuario. La figura 27B presenta una mancha elíptica de color blanco al centro del recuadro, lo cual es atribuible a las reflexiones internas ocasionadas por el sistema de iluminación. Dado que el contorno de la mancha central presente en la figura 27B no está bien definido, el algoritmo interpreta erróneamente estas irregularidades en color como la presencia de peces. Esto se hace evidente en la figura 27C en donde el algoritmo acota en pequeños rectángulos verdes lo que considera como peces, lo cual provocaría errores significativos en el proceso de conteo. Tal como se observó en la figura 26C que utiliza el color amarillo de fondo, cuando en el acuario no hay peces, el algoritmo genera una imagen completa en color blanco lo que deriva en consecuencia que el conteo sea un cero absoluto de peces presentes, lo cual es acorde a la situación real experimental planteada.

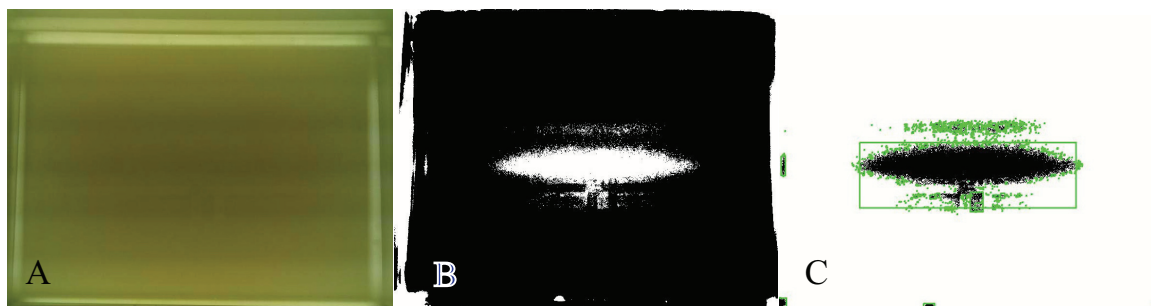


Figura 27: Visualización del procesamiento de una imagen del acuario sin peces de fondo color blanco. A) Imagen de entrada. B) Imagen binarizada. C) Imagen complementada.

Acorde a las variables presentes en un proceso real de conteo de peces vivos y con la finalidad de evaluar el entrenamiento (calibración) del algoritmo desarrollado para delimitar sus parámetros de operación, en las tablas III y IV se presentan los resultados de promediar para cada caso 16 imágenes capturadas del acuario sin peces (figura 26A y 27A), bajo diferentes condiciones de iluminación, niveles de agua y diferentes colores de fondo, siendo la tabla III la referente a los valores obtenidos en el primer sistema el cual esta basado en una computadora con Matlab y la tabla IV la referente a los valores obtenidos en el segundo sistema el cual esta basado en la Raspberry Pi empleando GNU Octave y la Raspi Cam.

En concordancia con lo planteado en la selección del color de fondo y a lo evidenciado en las figuras 26 y 27, el algoritmo responde de diferente manera al color de fondo, a la cantidad de agua presente en el acuario y a las condiciones de iluminación. En las tablas III y IV, se puede observar que el color amarillo huevo en el fondo del acuario junto con el sistema de iluminación operando, permite al algoritmo procesar de manera óptima y confiable la imagen de entrada, que en este caso al ser una imagen sin peces se obtiene un resultado de conteo con cero peces, observándose que es necesaria la iluminación artificial debido a que los peces generan sombras entre ellos mismos.

Tabla III: Calibración del algoritmo utilizado en el primer sistema basado en PC con Matlab: Se procesan y promedian 16 imágenes sin peces para cada caso con diferentes colores de fondo y distintas condiciones de iluminación.

Color de fondo	Conteo: imagen sin peces y dos litros de agua				Conteo: imagen sin peces y cuatro litros de agua			
	Con luces y flash	Solo luces	Solo flash	Sin luces ni flash	Con luces y flash	Solo luces	Solo flash	Sin luces ni flash
Amarillo huevo	0	0	0	2	0	0	0	0
Amarillo canario	0	0	3	2	0	0	0	0
Rosa	1	1	1	1	1	1	1	1
Naranja	27	14	231	1	15	22	22	1
Café	56	36	523	613	68	102	609	651
Azul rey	76	56	376	575	50	19	454	751
Blanco	83	98	73	91	92	1	60	71
Morado	122	25	195	145	108	41	178	204
Verde	198	271	119	104	234	233	85	94
Azul imperial	199	1	175	1	203	8	140	240

Tabla IV: Calibración del algoritmo utilizado en el segundo sistema basado en Raspberry Pi con GNU Octave: Se procesan y promedian 16 imágenes sin peces para cada caso con diferentes colores de fondo y distintas condiciones de iluminación.

Color de fondo	Conteo: Imagen sin peces y 2 litros de agua		Conteo: Imagen sin peces y 4 litros de agua	
	Con luces	Sin luces	Con luces	Sin luces
Amarillo canario	0	140	0	186
Amarillo huevo	0	0	0	0
Rosa	3	118	2	64
Naranja	25	42	11	193
Café	32	74	38	236
Azul rey	4	166	13	559
Blanco	94	70	150	92
Morado	7	248	12	245
Verde	64	0	20	0
Azul imperial	10	160	9	158

### III.4 Integración del sistema propuesto

El sistema contador de peces propuesto en este trabajo de tesis doctoral, se compone por los siguientes elementos:

- *Estructura metálica.*
- *Sub-sistema de iluminación.*
- *Sub-sistema de adquisición de imágenes.*
- *Computadora o sistema embebido.*
- *Algoritmo de procesamiento propuesto.*
- *Teclado y ratón.*
- *Monitor para el despliegue de resultados.*

La figura 28 muestra el primer sistema automático de conteo de peces desarrollado en este trabajo de tesis basado en el diagrama de bloques de la figura 13. Se trabaja con el software matemático llamado Matlab versión R2014a, siendo ejecutado en una computadora personal MacBook Pro con un procesador de 2.5 GHz Intel core i5, con una memoria RAM de 8 GB 1600 MHz DDR3 y un disco duro de 500 GB, las imágenes son adquiridas por la cámara marca Sony Cyber-shot modelo DSC-W180 con resolución de 10.1 Megapíxeles con una conexión USB, siendo en la interfaz gráfica de la computadora, donde se ven reflejados los resultados del procesamiento de las imágenes de entrada y del conteo de peces. El sistema de iluminación consiste de dos lámparas fluorescentes de policarbonato FLC-16W/65 de 16 W @ 110V<sub>ac</sub> cada una y cuya iluminación se distribuye uniformemente sobre la superficie del acuario a través del difusor de acrílico. Por cada imagen capturada de los peces alojados en el acuario, se despliega en la

interfaz la cantidad de peces presentes. El proceso se puede realizar de forma iterativa colocando nuevos peces, ya que de manera acumulativa al finalizar un proceso de conteo el sistema despliega la cantidad total de peces encontrados. El tiempo de conteo de peces por imagen es de aproximadamente 6 ms, lo que corresponde, al tiempo que transcurre desde el momento en que se captura la imagen y se muestra el resultado en la interfaz gráfica.

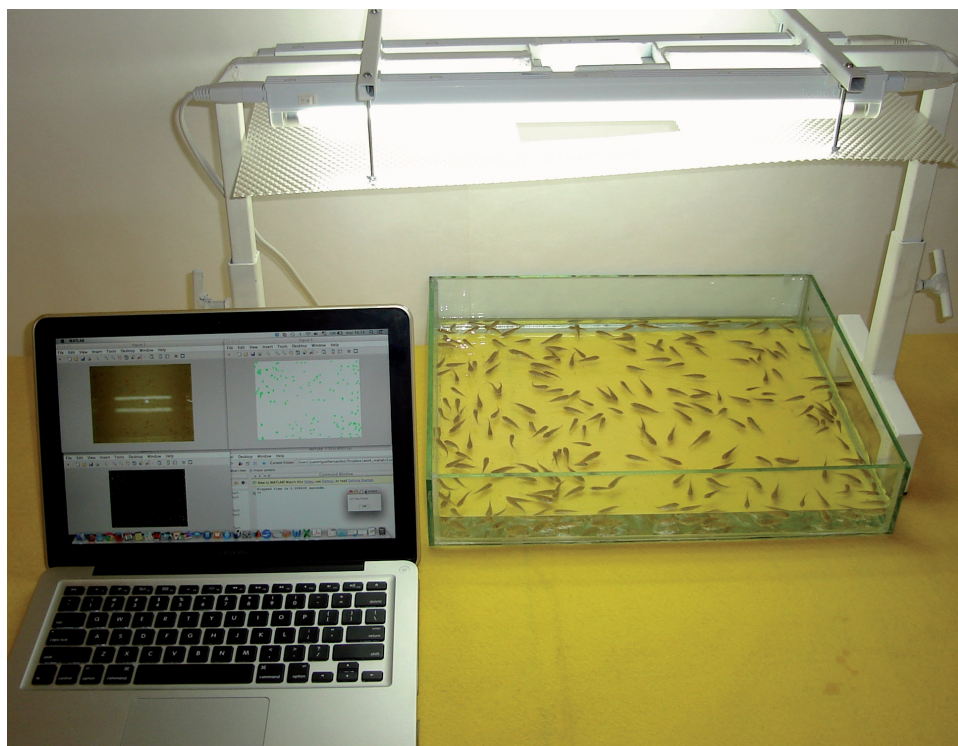


Figura 28: Arreglo experimental del primer sistema automático contador de peces basado en procesamiento digital de imágenes y empleando una computadora personal (PC).

La figura 29 muestra el segundo sistema automático desarrollado para el conteo de peces basado en el diagrama de bloques de la figura 15. Se trabaja con el software matemático llamado GNU Octave, siendo ejecutado en una Raspberry Pi 2 modelo B la cual basa su diseño en un SoC (System on a Chip) Broadcom BCM2836 el cual incluye un procesador Quad Core ARM Cortex A7 de 900 MHz, un GPU VideoCore IV y

una memoria SDRAM de 1 GB. El sistema de iluminación consiste de cuatro lámparas fluorescentes de policarbonato, dos de ellas son FLC-16W/65 de 16 W @ 110  $V_{ac}$  cada una y las otras dos son FCL-4006 de 8W @ 110 $V_{ac}$ .

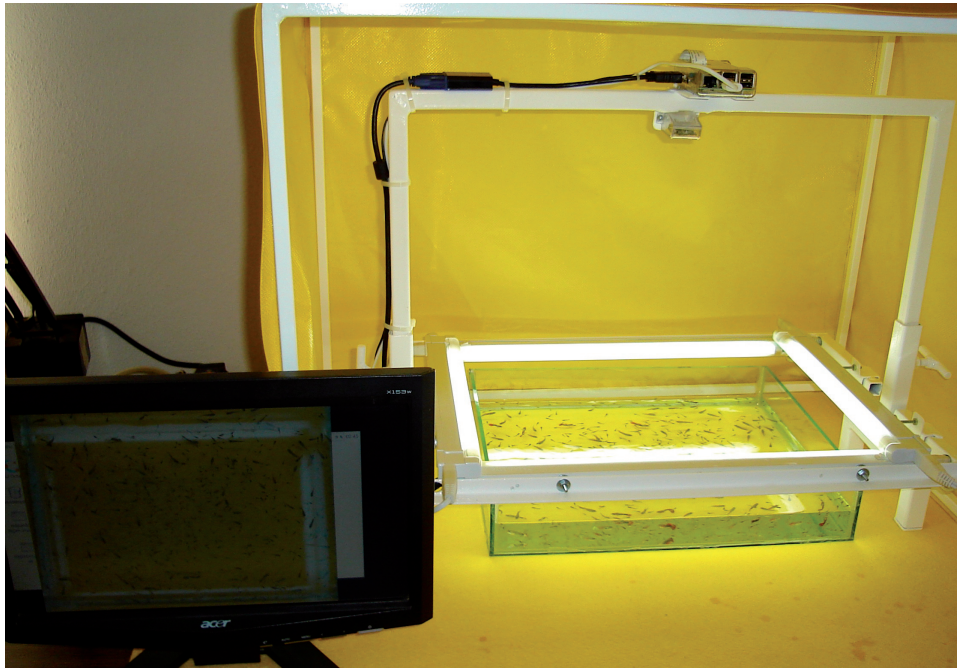


Figura 29: Arreglo experimental del segundo sistema automático contador de peces basado en procesamiento digital de imágenes y empleando el sistema embebido Raspberry Pi 2.

### III.5 Perspectiva de implementación en una granja de peces

La figura 30 muestra una ilustración del sistema propuesto dentro del escenario de una granja piscícola del mundo real. Los principales elementos hidráulicos, mecánicos y eléctricos que conforman este escenario son: Dos estanques grandes con capacidad de 25,000 litros de agua, una estructura metálica que soporta al contador de peces y se encuentra localizada entre los dos estanques. Una bomba centrífuga de 1 Hp @  $220V_{ac}$  para re-circulación forzada del agua, dos electro-válvulas para el control del flujo de agua. Tres válvulas check para evitar el retorno del agua y mantener un flujo unidireccional. También es necesario, un filtro de polipropileno, tres llaves de paso, una válvula de fondo y una rejilla, tubería de PVC 1 y 1.25". En el lado izquierdo de la figura 4.6, se encuentra el estanque de cultivo de peces, siendo este el lugar de crianza y engorda de los miles de peces. Cuando es necesario iniciar la contabilización de peces, el sistema embebido propuesto enciende las electro-válvulas y la bomba centrífuga propiciando un flujo de agua y peces hacia el acuario. El encendido/apagado de las electro-válvulas y bomba es mediante el uso de una etapa de potencia eléctrica interconectada a los pines GPIO (General Purpose Input/Output) del sistema embebido propuesto. Una vez que el sistema propuesto detecta la llegada de los primeros peces, se cierran las electro-válvulas para evitar la evacuación de los peces y se da inicio con el proceso de conteo de peces mediante el algoritmo propuesto. Cuando el proceso de conteo de peces en el acuario ha finalizado, son encendidas la electro-válvula de salida y la bomba de re-circulación para mover los peces desde el acuario hasta el estanque colector localizado en el lado derecho de la figura 4.6. Este proceso es repetido periódicamente hasta que es obtenida la cantidad de peces deseada. Además, el sistema de re-circulación forzada contempla el uso de filtros para mantener el agua libre de sedimentos. Las tres llaves de paso son para el mantenimiento del sistema y el remplazo de partes. La válvula de

fondo y la rejilla son instaladas para prevenir la succión de peces por la bomba.

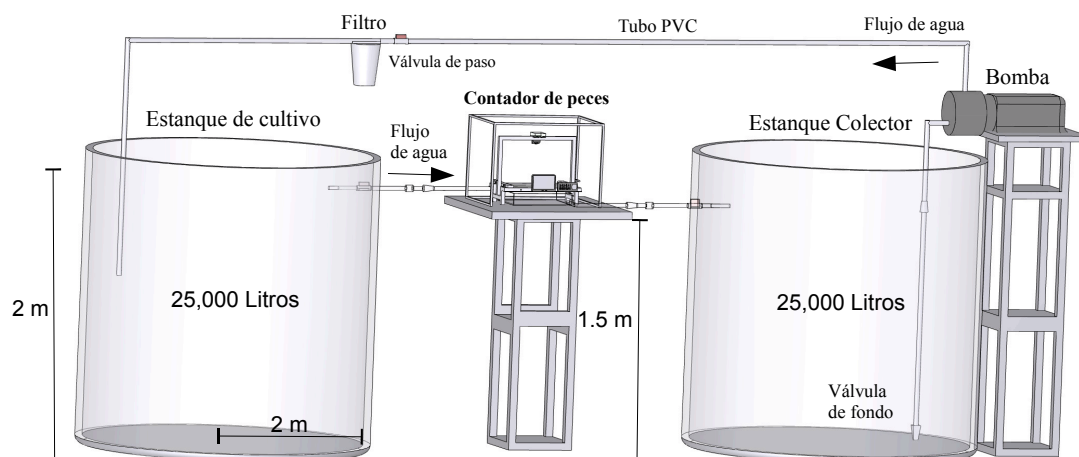


Figura 30: Implementación del sistema en una granja de peces.

# Capítulo IV

## Resultados experimentales

En esta sección se presentan los resultados obtenidos, mostrando en primer instancia lo obtenido utilizando los momentos invariantes de Hu, en donde se identifican y cuentan 5 especies de peces diferentes, posteriormente y mediante el uso de Matlab se contabilizan peces vivos dentro de una pecera y se presentan sus resultados, finalmente se cuentan peces vivos mediante el uso del sistema embebido Raspberry Pi 2 y usando la plataforma GNU Octave.

### IV.1 Resultados utilizando momentos invariantes de Hu (primer algoritmo propuesto)

En la figura 31 se presenta la imagen de entrada a procesar, donde el algoritmo propuesto identifica y contabiliza los peces presentes. Esta imagen es ideal, ya que no cuenta con ruido estadístico de ningún tipo, contiene 70 peces de distintos tamaños distribuidos aleatoriamente y los cuales están agrupados en 5 especies de la siguiente manera: 16 Bagres, 17 Besugos, 17 Sierras, 5 Tilapias y 15 Totoabas.

La figura 32A muestra la imagen principal de la figura 31 en blanco y negro, la cual se obtiene utilizando el método de (Otsu, 1975) y que corresponde al paso inicial para realizar el procesado de la imagen. La figura 32B muestra la identificación individual de cada pez acotándolo con un rectángulo verde. Una vez que el algoritmo identifica a un pez, lo muestra en pantalla de forma individual y calcula sus siete momentos de Hu. La figura 33 muestra un ejemplo de la extracción individual de los peces de la imagen original mostrada en la figura 31.



Figura 31: Imagen con 70 peces de 5 especies diferentes, ordenadas aleatoriamente y de diferentes tamaños.

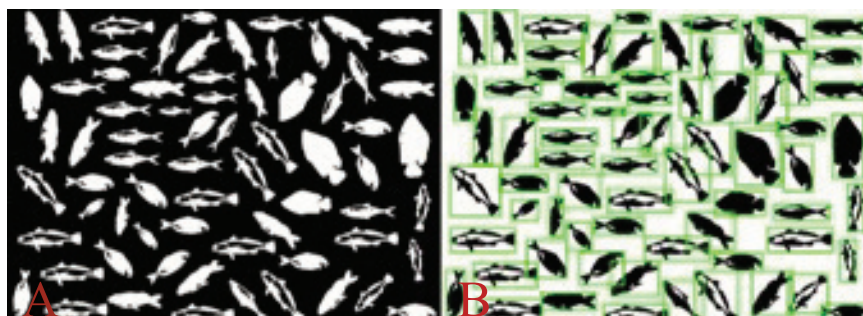


Figura 32: A) Imagen original binarizada, B) Identificación de cada pez.



Figura 33: Ejemplo de extracción individual de los peces de la imagen original empleando momentos de Hu.

La figura 34 muestra el resultado que entrega el primer algoritmo propuesto, en donde se muestra la cantidad de peces encontrados y el tipo de especie al que corresponde. Acorde con la distribución original de peces definida en la figura 31, el programa identifica y contabiliza correctamente los 70 peces, obteniendo una exactitud del 100% en el conteo e identificación de los peces.

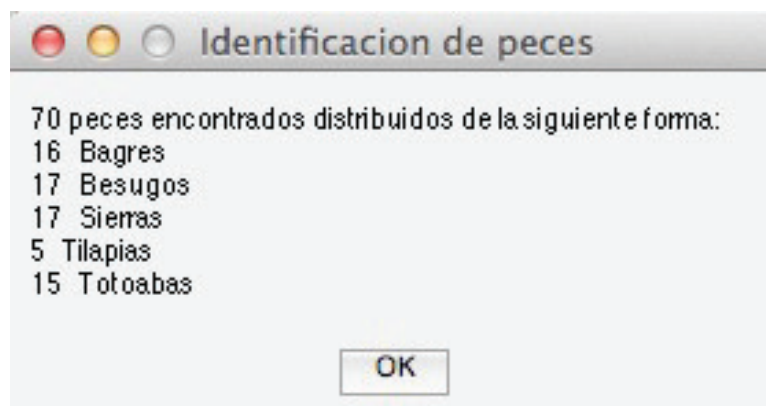


Figura 34: Resultado de la identificación y contabilización de peces.

Los momentos invariantes de Hu de la plantilla (capítulo III.3.1 figura 24) se muestran en la tabla V, en donde la primer columna es referente al nombre de los diferentes peces usados (Bagre, Besugo, Sierra, Tilapia y Totoaba), la segunda columna contiene los momentos  $\phi_1$  dando resultado promedio de 6.701, la tercer columna muestra los momentos  $\phi_2$  con resultado promedio de 14.22, la cuarta columna muestra los momentos  $\phi_3$  con resultado promedio de 26.73, la quinta columna muestra los momentos  $\phi_4$  con resultado promedio de 28.01, la sexta columna muestra los momentos  $\phi_5$  con resultado promedio de 55.61, la séptima columna muestra los momentos  $\phi_6$  con resultado promedio de 35.81 y finalmente la octava columna muestra los momentos  $\phi_7$  con resultado promedio de 56.13.

Tabla V: Momentos de Hu de los peces a identificar.

Nombre del pez	Momentos Invariantes de Hu						
	$\phi_1$	$\phi_2$	$\phi_3$	$\phi_4$	$\phi_5$	$\phi_6$	$\phi_7$
Bagre	6.57491	13.77810	27.90019	27.46714	55.38467	34.40242	55.67111
Besugo	6.71676	14.22185	26.18038	28.62795	56.24992	35.88055	56.89339
Sierra	6.67338	14.06162	25.94435	27.56513	54.35811	35.48367	55.94939
Tilapia	6.95661	15.28206	27.27119	28.79226	57.23066	38.61921	57.13427
Totoaba	6.58157	13.78056	26.34122	27.57500	54.83895	34.67420	55.01523
<b>Promedio</b>	<b>6.70</b>	<b>14.22</b>	<b>26.73</b>	<b>28.01</b>	<b>55.61</b>	<b>35.81</b>	<b>56.13</b>

La figura 35 muestra la imagen original (figura 31) alterada con 4% de ruido del tipo sal y pimienta para la identificación y conteo de peces bajo el primer algoritmo propuesto. La imagen original se somete a estas condiciones de ruido para estimar cuanto afecta en el conteo final de peces. Bajo estas condiciones de ruido en la imagen, se obtiene una exactitud del 95% (Hernández-Ontiveros J. M., 2014).



Figura 35: Imagen original con 70 peces y con 4% de ruido del tipo sal y pimienta.

La figura 36 muestra la figura original (figura 31) alterada con 4% de ruido gaussiano, media 0 y varianza 0.04, para la identificación y conteo de peces bajo el algoritmo propuesto. Con estas condiciones de ruido, se observa que el ruido nuevamente afecta en forma mínima al conteo final de peces y presenta una ligera desviación en la identificación. Bajo estas condiciones de ruido en la imagen, se obtiene una exactitud del 92% (Hernández-Ontiveros J. M., 2014).



Figura 36: Imagen original con 70 peces y con 4% de ruido gaussiano, media 0 y varianza 0.04.

Usando los primeros tres momentos de Hu para cada una de las especies de la plantilla base (capítulo III.3.1 figura 24) y mostrados en la tabla V, se construye una gráfica 3D mostrada en la figura 37 que ilustra la clasificación de los peces a identificar. La figura 37 muestra además la distribución de los 70 peces mostrados en la figura 35, alterada con ruido del tipo sal y pimienta. La figura 37 hace evidente varios traslapes entre los momentos, particularmente entre las especies del pez Sierra y el Tilapia. Esto da lugar a las diferentes desviaciones que se presentan en el proceso de identificación.

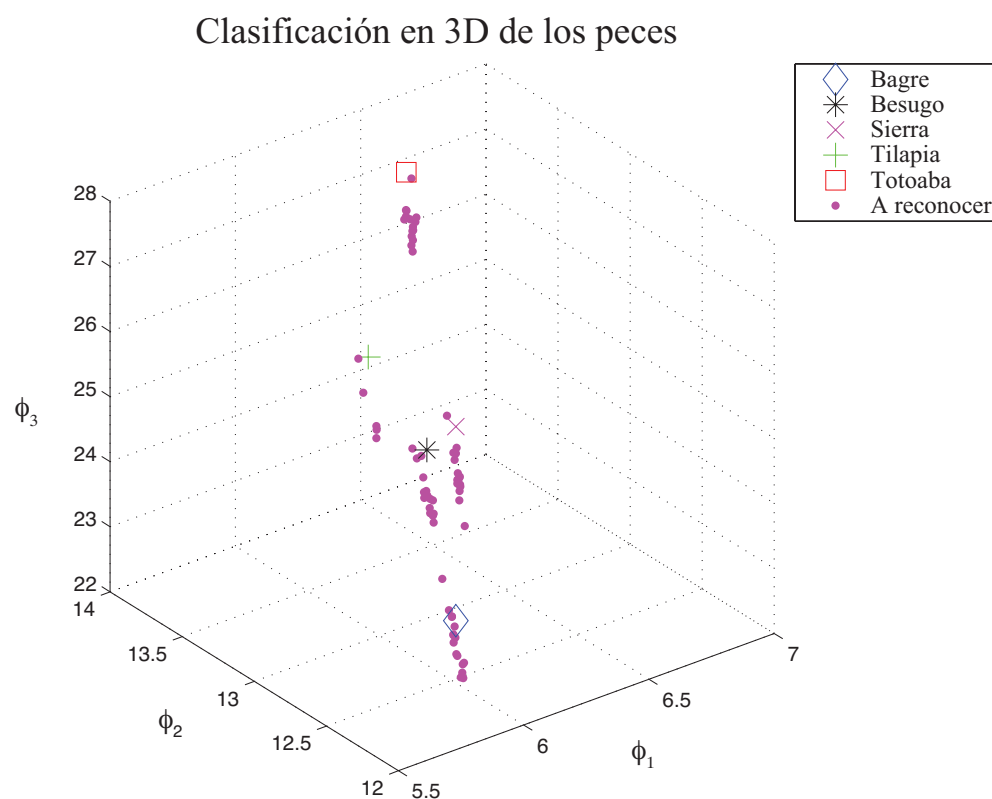


Figura 37: Clasificación en 3D de peces a identificar bajo ruido del tipo sal y pimienta.

Tabla VI: Validación del primer algoritmo tomando en cuenta ruido sal y pimienta y ruido gaussiano.

Cantidad de peces en imagen original	Aplicando Ruido Sal y Pimienta		Aplicando Ruido Gaussiano	
	Peces identificados correctamente	Exactitud (%)	Peces identificados correctamente	Exactitud (%)
30	28	93.33	30	100
32	30	93.75	30	93.75
34	32	94.12	32	94.12
36	34	94.44	32	88.89
38	36	94.73	32	84.21
40	38	95.00	36	90.00
42	40	95.24	38	90.48
44	42	95.45	40	90.91
46	44	95.65	42	91.30
48	46	95.83	44	91.67
50	48	96.00	46	92.00
52	48	92.31	48	92.31
54	52	96.30	50	92.59
56	52	92.86	52	92.86
58	56	96.55	54	93.10
60	58	96.67	56	93.33
62	60	96.77	58	93.55
64	62	96.88	60	93.75
66	64	96.97	62	93.94
68	66	97.06	64	94.12
70	66	94.29	66	94.29
	<b>Exactitud Promedio</b>	<b>95.25</b>	<b>Exactitud Promedio</b>	<b>92.44</b>

La tabla VI presenta varias pruebas de validación para el algoritmo propuesto. Para cada una de las pruebas de validación, la imagen original a procesar como la mostrada en la figura 31, es diferente. Se cambian: i) la cantidad total de peces presentes en la imagen con un rango de variación de 30 a 70 y ii) su distribución es aleatoria. Esto se muestra en la primera columna de la tabla VI. Cada imagen es alterada con ruido tanto del tipo sal y pimienta, como del tipo gaussiano. En la tabla VI se muestra la cantidad total de peces identificados y contabilizados correctamente. Para cada análisis se presenta el porcentaje estimado de la exactitud del proceso de identificación. Se observa que cuando la imagen original es alterada bajo ruido del tipo sal y pimienta, se obtiene una exactitud promedio de 95.25% en la etapa de identificación por especie, mientras que cuando se altera con ruido del tipo gaussiano, se obtiene una exactitud promedio de 92.44%. La cantidad total de peces presentes en las imágenes originales bajo análisis, no presenta ninguna desviación independientemente del tipo de ruido a la que es expuesta.

## **IV.2 Resultados del segundo algoritmo propuesto empleando una PC con Matlab**

En la figura 38, se muestran los resultados del procesamiento digital de una imagen de entrada correspondiente al acuario con peces vivos. Los peces corresponden a la familia de los Platys Mickey Mouse (*Xiphophorus maculatus*) con una talla promedio de 1.2 cm. La figura 38A corresponde a la imagen de entrada donde se observa el comportamiento y movilidad aleatoria de los peces. La figura 38B muestra el resultado de binarizar la imagen de entrada y en la figura 38C se muestra la identificación de los peces presentes acotándolos en un rectángulo verde para finalmente contabilizarse empleando el segundo algoritmo propuesto y usando el método de conteo de (Huang y Wang, 2009) y (Fan y Liu, 2013) visto en el capítulo II.3.



Figura 38: Resultados del procesamiento digital de la imagen de entrada usando Matlab: A) Imagen de entrada con 70 peces, B) Imagen binaria, C) Identificación de los peces.

En la tabla VII, se presentan los resultados del conteo de peces a partir de una sola imagen capturada para diferentes cantidades de Guppies (*Poecilia Reticulata*) con una talla promedio de 5 mm. En la primer columna se indica el tipo de pez a contabilizar, en la segunda columna la cantidad real de peces presentes en el acuario, en la tercer columna se indica los peces contabilizados por el sistema, en la cuarta columna se muestra el tiempo de procesamiento transcurrido durante el conteo y en la quinta columna la exactitud en donde se muestra que el promedio de exactitud del conteo es de 95.65%.

De la tabla VII se infiere el error que se obtiene del conteo final por parte del contador usando el segundo algoritmo propuesto, a partir de procesar una sola imagen. Un análisis cuidadoso de las causas de estas diferencias se debe a dos posibles situaciones: i) nivel de resolución de la cámara y ii) falta de iluminación uniforme. Una limitación de este sistema, se presenta cuando los peces tienen una talla menor a los 5 mm, el sistema no es capaz de identificar información, por lo que, en el proceso de convertir a binaria la imagen no es posible detectar una magnitud susceptible a contabilizarse. Sin embargo, esta limitación se puede mejorar, incrementando la resolución de la cámara. Por otra parte, la falta de una iluminación uniforme sobre la superficie del acuario hace

presente el reflejo de las lámparas y esto puede generar el incremento o decremento de la cantidad de peces presentes de manera aleatoria. Pudiera darse el caso que en el instante de tomar una fotografía uno o más peces se ubiquen justo en la zona de mayor reflejo (saturación) de alguna de las lámparas (figura 39A), lo que el algoritmo no es capaz de detectar por lo que deriva en un decremento de la cantidad de peces contabilizados. Por otra parte, cuando la especie tiene una talla mayor de 10 mm y atraviesa el reflejo de una de las lámparas (figura 39D), el pez es aparentemente dividido lo que el algoritmo interpreta como dos peces, por lo que deriva en un incremento en la cantidad de peces contabilizados. La figura 39, muestra las posibles situaciones susceptibles a propiciar estos errores en el conteo.

Tabla VII: Conteo de peces de la especie Guppies (*Poecilia reticulata*) con un tamaño de 5 mm usando el sistema contador automático basado en PC y Matlab (segundo algoritmo propuesto).

Especie	Cantidad real de peces en el acuario	Peces contabilizados por el sistema	Tiempo de procesamiento del sistema (ms)	Exactitud del sistema (%)
Guppies ( <i>Poecilia reticulata</i> ) Tamaño de 5 mm	130	127	8.28	97.69
	120	111	7.38	92.50
	110	104	6.85	94.54
	100	95	6.50	95.00
	90	87	8.20	96.67
	80	78	5.30	97.50
			<b>Promedio</b>	<b>95.65</b>

En la figura 39A se observan claramente 3 peces, uno de ellos se encuentra fuera del reflejo de una de las lámparas, otro se encuentra parcialmente bajo el reflejo y el tercero totalmente dentro de la zona de reflejo de luz. Se observa que el algoritmo al binarizar esta imagen (figura 39B), elimina al pez que se encuentra totalmente en la zona de reflejo de la lámpara. En consecuencia, el valor final que estima el algoritmo es de 2 peces identificados (figura 39C) en lugar de los tres peces inicialmente presentes

en la imagen original (figura 39A). Por otra parte, la figura 39D muestra solamente un pez atravesando la zona de reflejo de una de las lámparas de luz. En la figura 39E correspondiente a la imagen binaria de la imagen de entrada 39D, se aprecia como el pez es partido por la mitad, lo que deriva que el algoritmo identifique dos peces en lugar de uno (figura 39F).

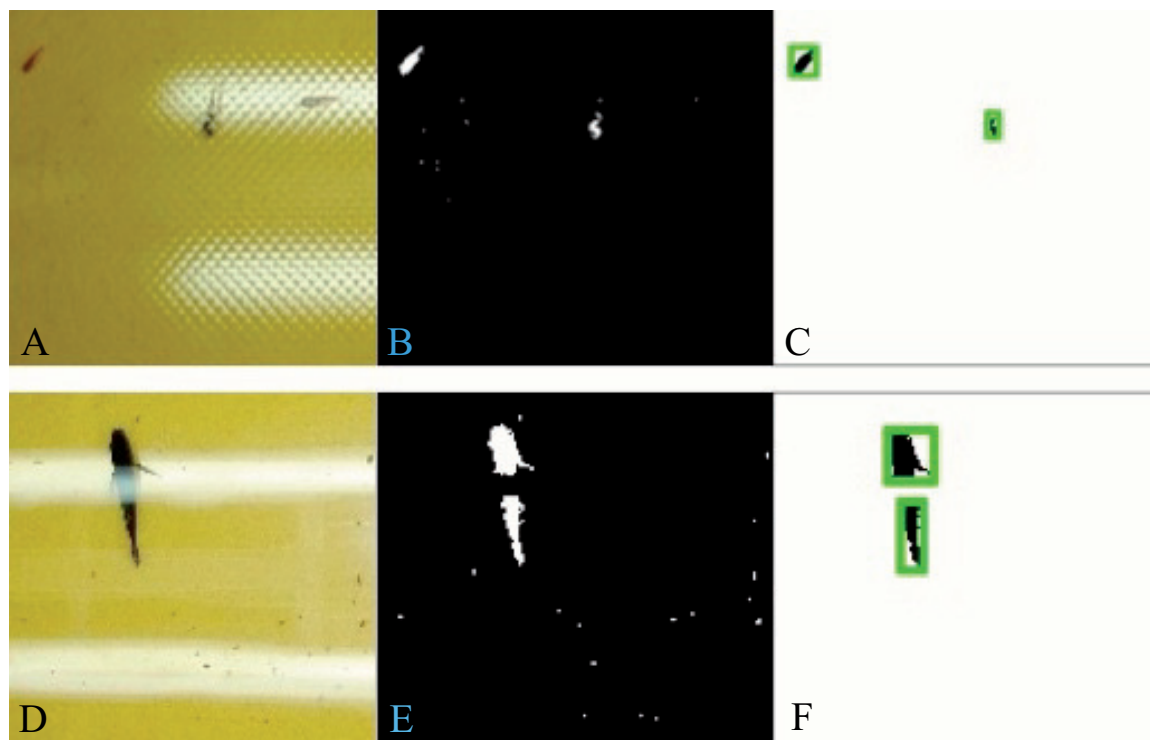


Figura 39: Situaciones aleatorias susceptibles a motivar errores en el conteo de peces.

En la tabla VIII se presentan los resultados de las pruebas de conteo de 115 peces vivos de diferentes especies y distintas tallas, a partir de una sola imagen tomada y empleando el segundo algoritmo propuesto. En la primera columna se presentan las diferentes especies de peces en estudio es decir: Guppies (*Poecilia reticulata*), Mollies (*Poecilia sphenops*), Platys Mickey Mouse (*Xiphophorus maculatus*) y Tilapia del Nilo (*Oreochromis niloticus*). En la segunda columna se indica el tamaño promedio de los peces, en la tercera columna se muestra el tiempo de procesamiento transcurrido para

el conteo automático y en la cuarta columna se indica el porcentaje de exactitud del instrumento desarrollado. En la tabla VIII, se puede observar que el rango de exactitud del sistema es de 96.29% a 97.87%, lo que significa que el conteo es confiable. El tiempo de procesamiento oscila entre 3.01 ms y 6.91 ms.

Tabla VIII: Pruebas de conteo de diferentes especies de peces y diferentes tallas, a partir de una sola imagen capturada de peces vivos, en todos los casos la cantidad real de peces fue de 115 usando PC y Matlab.

<b>Especie</b>	<b>Tamaño promedio (cm)</b>	<b>Tiempo de procesamiento del sistema (ms)</b>	<b>Exactitud del sistema (%)</b>
Guppies ( <i>Poecilia reticulata</i> )	0.5	6.914	96.29
Mollies ( <i>Poecilia sphenops</i> )	3.3	3.012	96.55
Platys Mickey ( <i>Xiphophorus maculatus</i> )	1.2	3.813	97.87
Tilapia del Nilo ( <i>Oreochromis niloticus</i> )	1.3	4.825	97.63
		<b>Promedio</b>	<b>97.08</b>

### IV.3 Resultados empleando Raspberry Pi con GNU Octave

Siguiendo el mismo proceso de conteo de peces que se utiliza con una PC y Matlab, al realizar la migración a Raspberry Pi 2, se decide trabajar con GNU Octave, debido a sus similitudes con Matlab, además de implementar el método de (Huang y Wang, 2009) para el conteo de peces descrito en el capítulo II.3. Se reproduce en la medida de lo posible los mismos experimentos por lo que en la figura 40, se muestran los resultados del procesamiento digital de una imagen de entrada correspondiente al acuario con peces vivos pero ahora usando la cámara digital de Raspberry Pi. Los peces corresponden a la familia de los Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm. La

figura 40A corresponde a la imagen de entrada original donde se puede observar el comportamiento y movilidad aleatoria de los peces. La figura 40B muestra el resultado de convertir a binaria la imagen de entrada y en la figura 40C se muestra la identificación de los peces presentes, acotándolos en un rectángulo verde para finalmente contabilizarse por el segundo algoritmo propuesto.

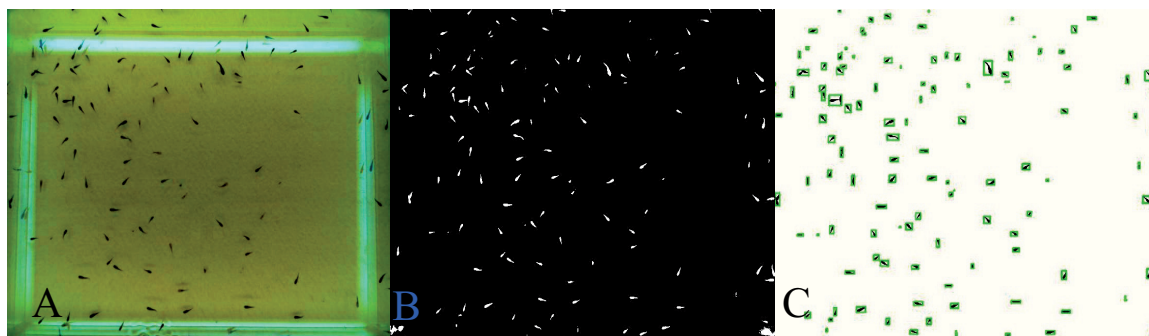


Figura 40: Resultados del procesamiento digital de la imagen de entrada usando Raspberry Pi: A) Imagen de entrada con 130 peces, B) Imagen binaria, C) Identificación de los contornos de los peces.

Dados los problemas encontrados en el primer método propuesto con una PC y Matlab, en este nuevo sistema se opta por mejorar la validación del instrumento y es en ese sentido se decide que para validar el instrumento es necesario realizar diferentes conteos, el primero es contar los peces de forma manual, una vez identificado este dato, se prosigue con un segundo conteo que consiste en utilizar el algoritmo propuesto realizando la captura de una secuencia de 16 imágenes para contabilizar los peces vivos con resultados confiables, el algoritmo captura una imagen cada 2 segundos y finalmente contabiliza los peces. La Figura 41 muestra las primeras 16 imágenes realizadas a 350 peces de la familia de los Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm.

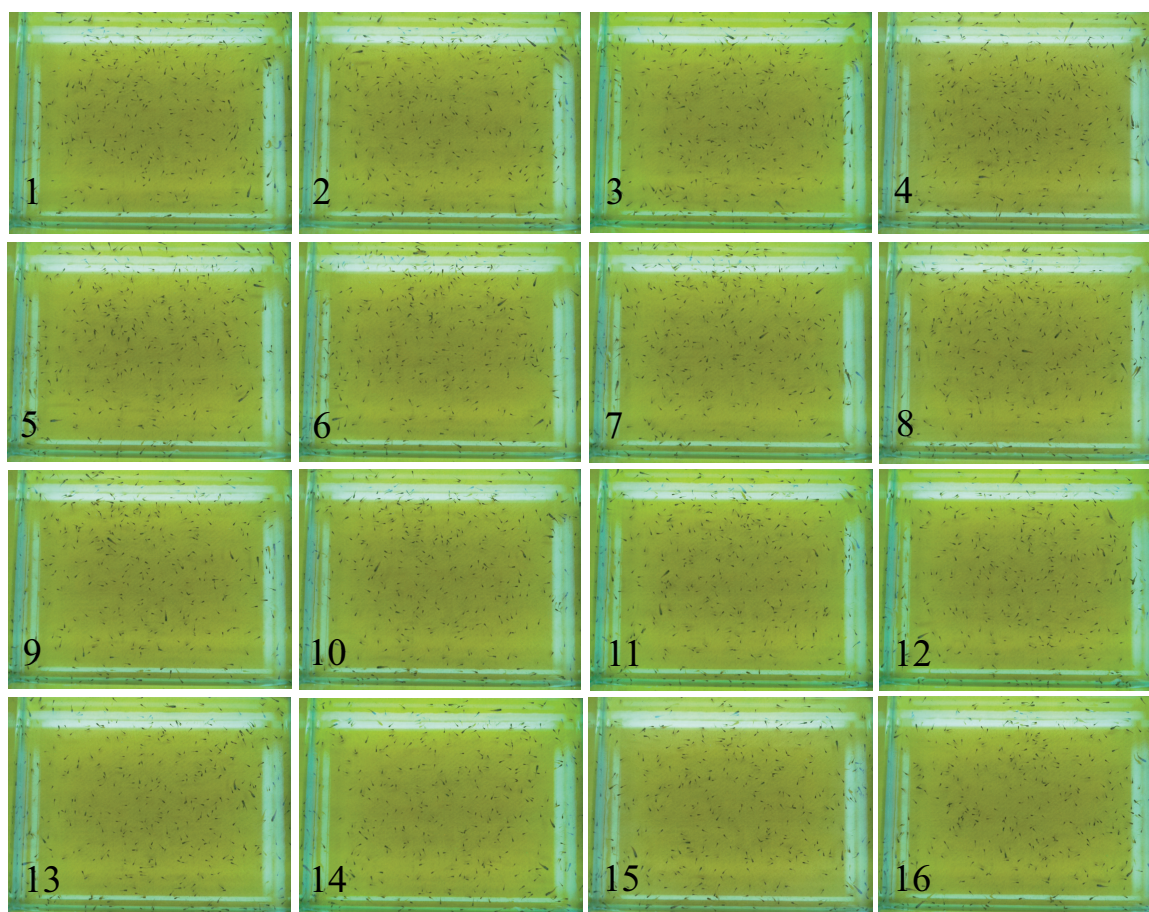


Figura 41: Secuencia de 16 imágenes capturadas con cámara de Raspberry Pi cada 2 segundos a la familia de Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm.

Debido al costo elevado de un sistema basado en PC con Matlab versus un sistema embebido tal como Raspberry Pi con código abierto, es necesario realizar los conteos de peces con ambas plataformas para evaluar sus desempeños en cuanto a su exactitud (AR), por lo que en las figuras 42 a 51 se muestran los resultados obtenidos con ambos sistemas, se puede observar que los resultados con Matlab y Raspberry Pi son similares, por lo que se concluye que el sistema embebido propuesto es confiable y de bajo costo.

La figura 42 muestra los resultados de contabilizar 350 Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave y el conteo con la PC usando Matlab son muy similares, el valor promedio contabilizado con PC es de 343.50, mientras que con Raspberry Pi el valor promedio es de 342.63, el porcentaje de exactitud con PC es de 98.14% mientras que con Raspberry Pi es de 97.89%.

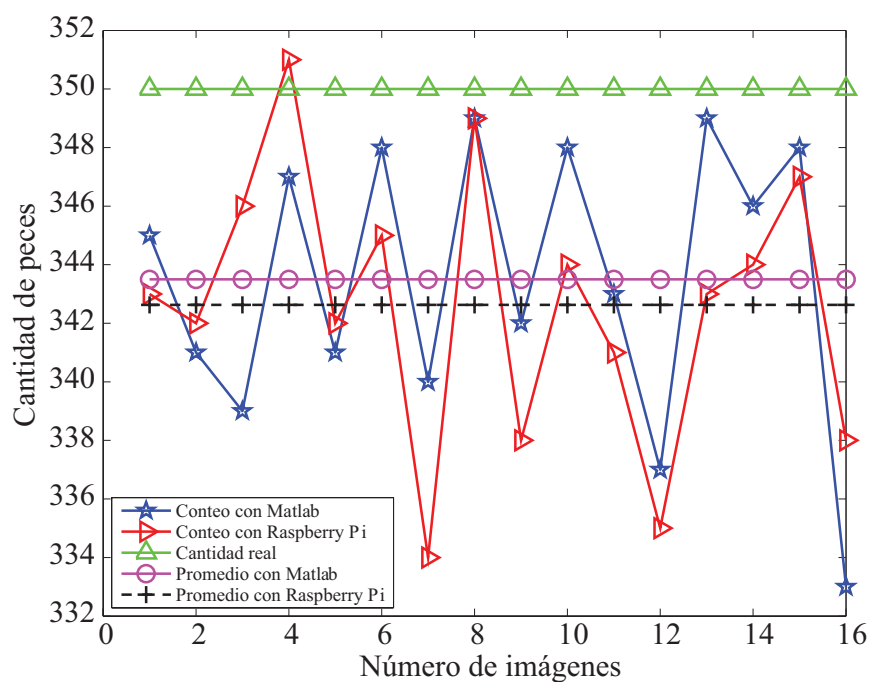


Figura 42: Resultados de contabilizar 350 Guppies (*Poecilia Reticulata*) con una talla promedio de 1.2 cm.

La figura 43 muestra los resultados de contabilizar 250 Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave y el conteo con la PC usando Matlab son muy similares, el valor promedio contabilizado con PC es de 243.19, mientras que con Raspberry Pi el valor promedio es de 242.56, el porcentaje de exactitud con PC es de 97.28% mientras que con Raspberry Pi es de 97.02%.

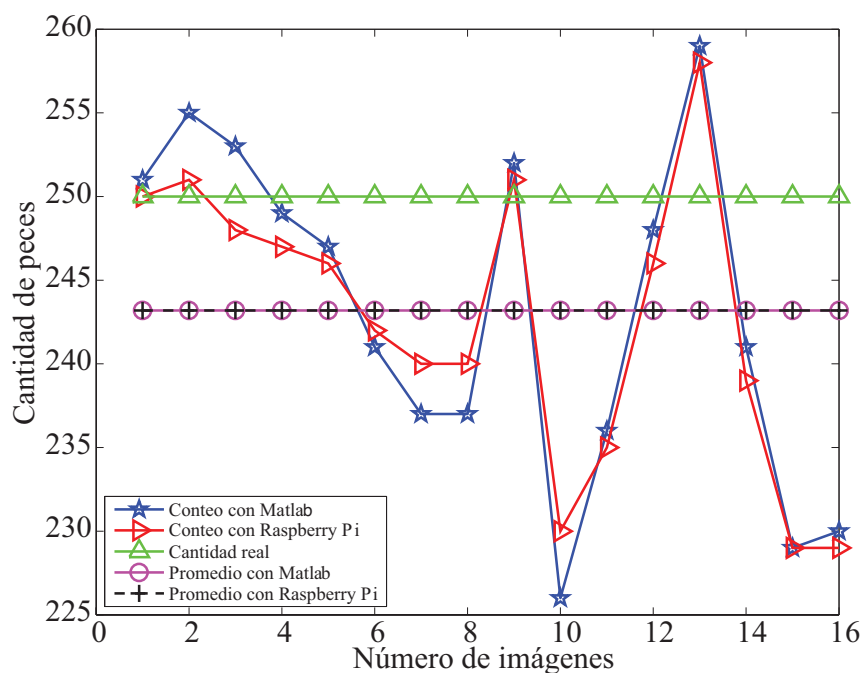


Figura 43: Resultados de contabilizar 250 Guppies (*Poecilia Reticulata*) con una talla promedio de 1.2 cm.

La figura 44 muestra los resultados de contabilizar 150 Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave y el conteo con la PC usando Matlab son casi iguales, el valor promedio contabilizado con PC es de 149.75, mientras que con Raspberry Pi el valor promedio es de 148.25, el porcentaje de exactitud con PC es de 99.83% mientras que con Raspberry Pi es de 98.83%.

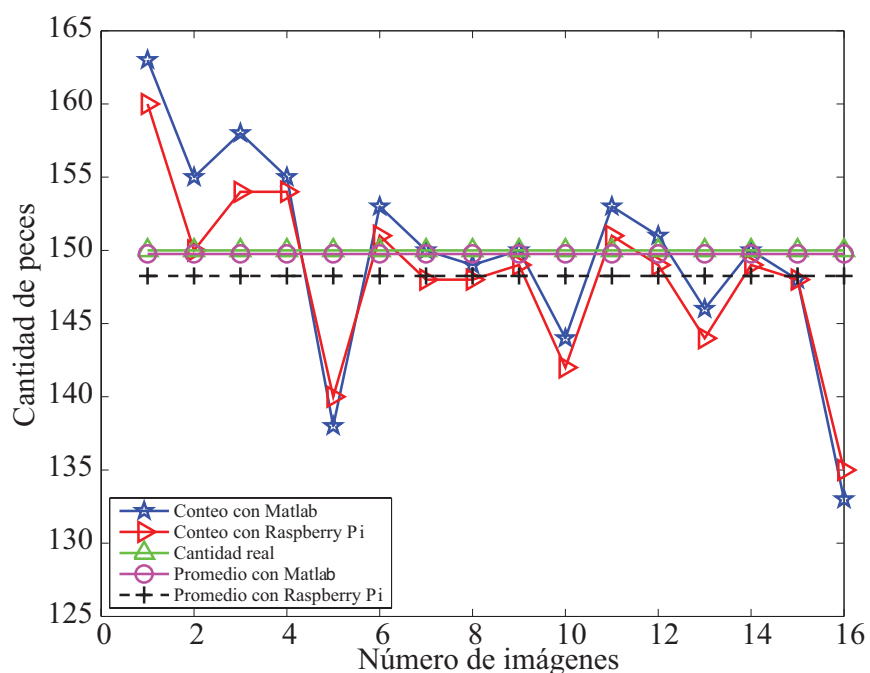


Figura 44: Resultados de contabilizar 150 Guppies (*Poecilia Reticulata*) con una talla promedio de 1.2 cm.

La figura 45 muestra los resultados de contabilizar 65 Guppies (*Poecilia reticulata*) con una talla promedio de 1.2 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave y el conteo con la PC usando Matlab nuevamente son muy similares, el valor promedio contabilizado con PC es de 62.06, mientras que con Raspberry Pi el valor promedio es de 64.00, el porcentaje de exactitud con PC es de 95.48% mientras que con Raspberry Pi es de 98.46%.

En la tabla IX, se presentan los resultados del conteo de peces a partir de 16 imágenes capturadas para diferentes cantidades de Guppies (*Poecilia Reticulata*) con una talla promedio de 1.2 cm y obteniendo un tiempo promedio de procesamiento del sistema por imagen de 230 segundos. En la primer columna se indica el tipo de especie, en la segunda se indica la cantidad real de peces presentes en el acuario, en la tercer columna se indica los peces contabilizados y en la cuarta columna se puede observar el porcentaje de exactitud del sistema. El promedio de exactitud del sistema es de 96.43%.

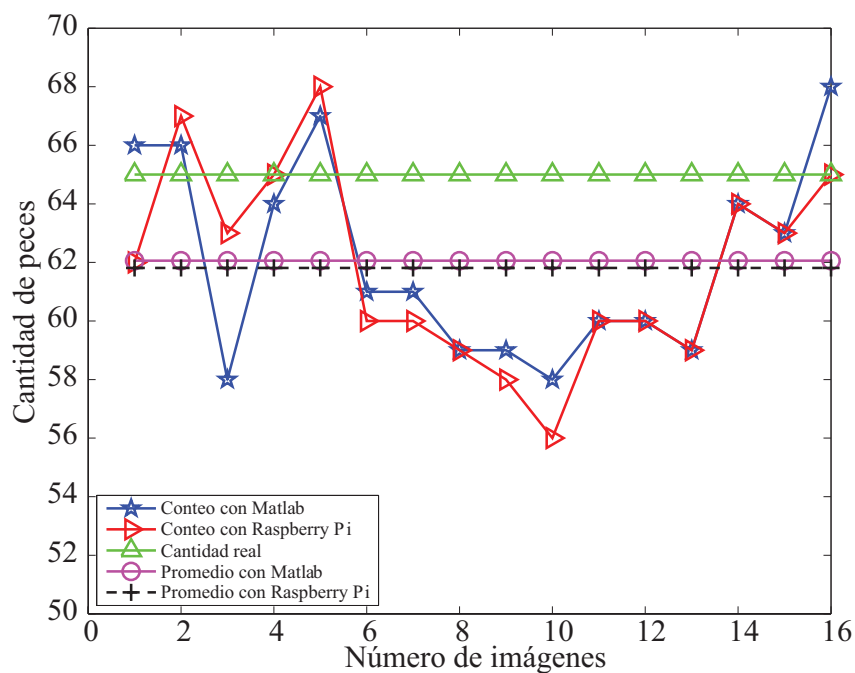


Figura 45: Resultados de contabilizar 65 Guppies (*Poecilia Reticulata*) con una talla promedio de 1.2 cm.

Tabla IX: Validación del conteo de peces de la especie Guppies (*Poecilia reticulata*) con un tamaño de 1.2 cm usando el sistema contador automático basado en Raspberry Pi.

Especie	Cantidad real de peces en el acuario	Peces contabilizados por el sistema propuesto	Exactitud del sistema (%)
Guppies ( <i>Poecilia reticulata</i> ) Tamaño de 1.2 cm.	350	342.63	97.89
	300	278.10	92.70
	250	242.56	97.02
	200	198.00	99.00
	150	148.25	98.83
	100	93.00	93.00
	65	64.00	98.46
	<b>Promedio</b>		<b>96.43</b>

Continuando con la presentación de resultados, se realizan conteos a la misma especie de Guppies pero de tamaño superior, es decir de 2.3 cm. La figura 46 muestra los resultados de contabilizar 195 Guppies (*Poecilia reticulata*) con una talla promedio de 2.3 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave se encuentra casi totalmente empalmado con el conteo con la PC usando Matlab, el valor promedio contabilizado con PC es de 189.31, mientras que con Raspberry Pi el valor promedio es de 190.13, el porcentaje de exactitud con PC es de 97.08% mientras que con Raspberry Pi es de 97.50%.

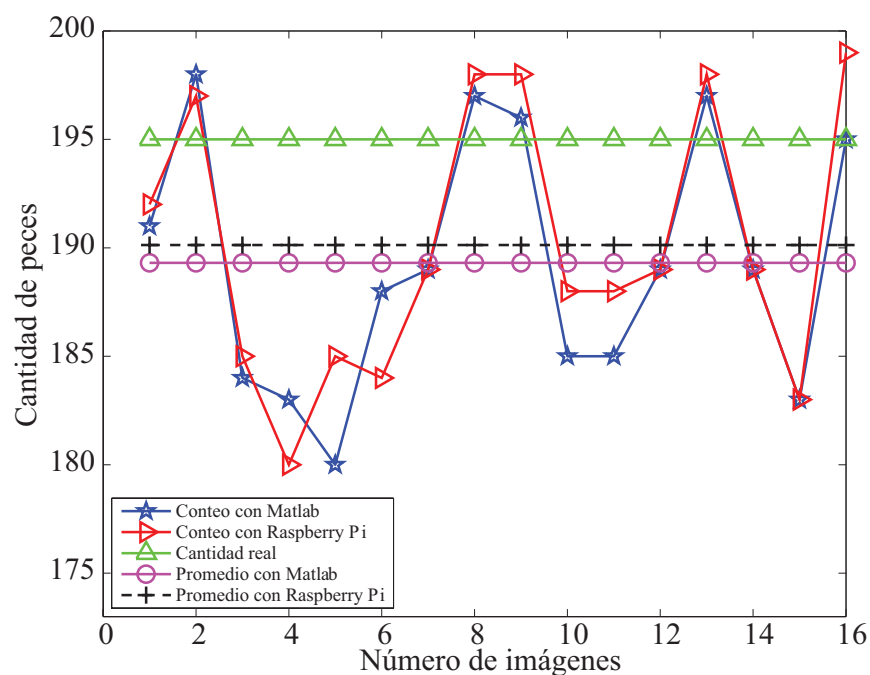


Figura 46: Resultados de contabilizar 195 Guppies (*Poecilia Reticulata*) con una talla promedio de 2.3 cm.

La figura 47 muestra los resultados de contabilizar 115 Guppies (*Poecilia reticulata*) con una talla promedio de 2.3 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave se encuentra nuevamente muy similar al conteo con la PC usando Matlab, el valor promedio contabilizado con PC es de 111.00, mientras que con Raspberry Pi el valor promedio es de 110.44, el porcentaje de exactitud con

PC es de 96.52% mientras que con Raspberry Pi es de 96.03%.

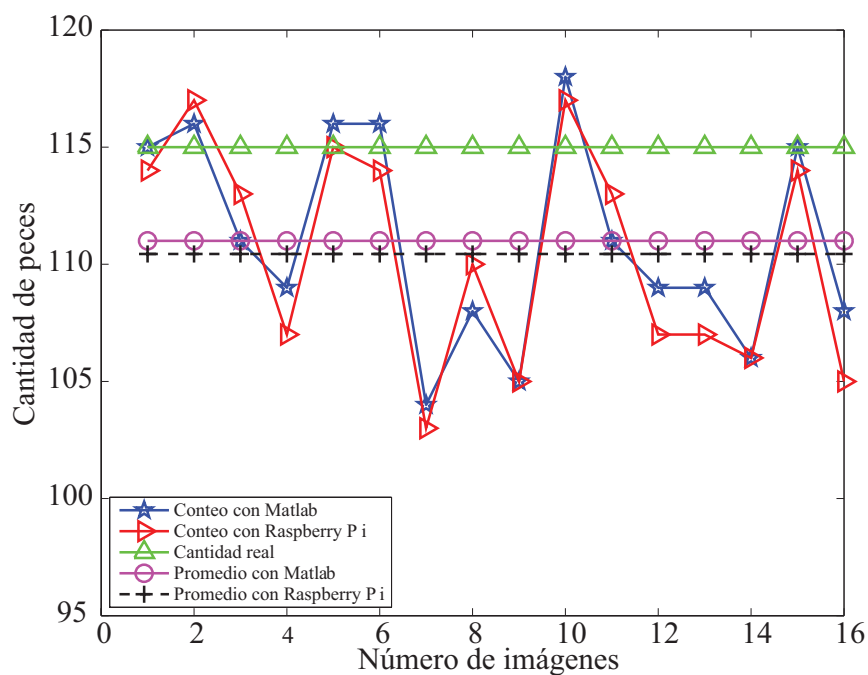


Figura 47: Resultados de contabilizar 115 Guppies (*Poecilia Reticulata*) con una talla promedio de 2.3 cm.

La figura 48 muestra los resultados de contabilizar 65 Guppies (*Poecilia reticulata*) con una talla promedio de 2.3 cm, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave se encuentra mínimamente por debajo del conteo con la PC usando Matlab, el valor promedio contabilizado con PC es de 63.69, mientras que con Raspberry Pi el valor promedio es de 60.25, el porcentaje de exactitud con PC es de 97.98% mientras que con Raspberry Pi es de 92.69%.

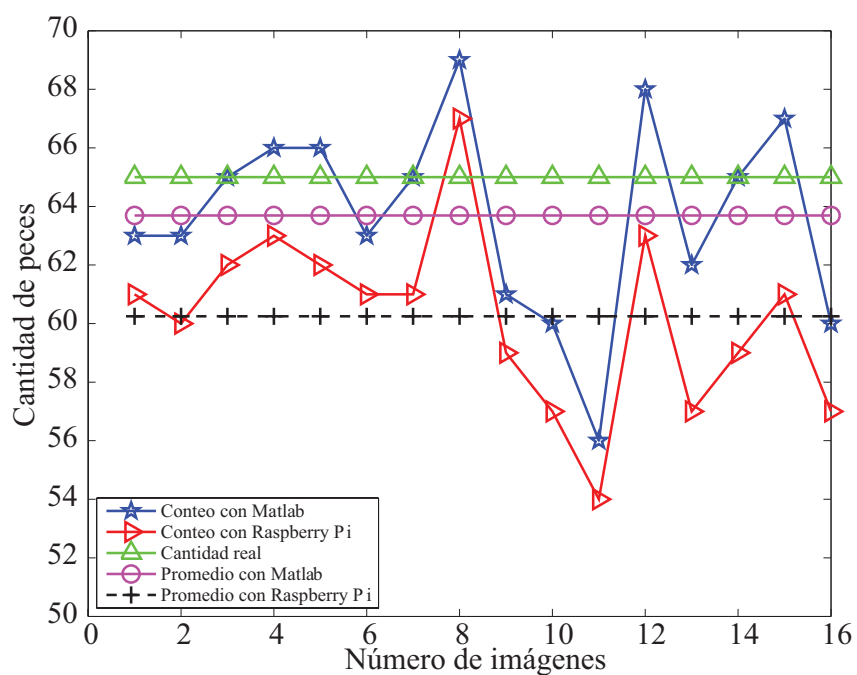


Figura 48: Resultados de contabilizar 65 Guppies (*Poecilia Reticulata*) con una talla promedio de 2.3 cm.

En la tabla X, se presentan los resultados del conteo de peces nuevamente a partir de 16 imágenes capturadas para diferentes cantidades de Guppies (*Poecilia Reticulata*) con una talla promedio de 2.3 cm y un tiempo promedio de procesamiento del sistema por imagen de 230 segundos. En la primer columna se indica el tipo de especie, en la segunda columna se muestra la cantidad real de peces presentes en el acuario, en la tercer columna se observan los peces contabilizados y en la cuarta columna se indica la exactitud del sistema. El promedio de exactitud del sistema con Raspberry Pi es de 93.69%.

Tabla X: Conteo de peces de la especie Guppies (*Poecilia reticulata*) con un tamaño de 2.3 cm usando el sistema contador automático con Raspberry Pi.

<b>Especie</b>	<b>Cantidad real de peces en el acuario</b>	<b>Peces contabilizados por el sistema propuesto Raspberry Pi</b>	<b>Exactitud del sistema (%)</b>
Guppies ( <i>Poecilia reticulata</i> ) Tamaño de 2.3 cm.	195	190.13	97.50
	155	140.00	90.32
	120	111.00	92.50
	115	110.44	96.03
	100	95.00	95.00
	85	78.00	91.76
	65	60.25	92.69
		<b>Promedio</b>	<b>93.69</b>

Finalmente se realizaron conteos con la especie denominada Mollies (*Poecilia Sphenops*), con un tamaño promedio de 0.5 cm. La figura 49 muestra los resultados de contabilizar 130 Mollies, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave se encuentra en ocasiones por arriba del conteo con la PC usando Matlab y en ocasiones por debajo, el valor promedio contabilizado con PC es de 124.06, mientras que con Raspberry Pi el valor promedio es de 122.88, el porcentaje de exactitud con PC es de 95.43% mientras que con Raspberry Pi es de 94.52%.

La figura 50 muestra los resultados de contabilizar 95 Mollies, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave se encuentra en ocasiones por arriba del conteo con la PC usando Matlab y en ocasiones por debajo, el valor promedio contabilizado con PC es de 91.44, mientras que con Raspberry Pi el valor promedio es de 91.88, el porcentaje de exactitud con PC es de 96.25% mientras que con Raspberry Pi es de 96.71%.

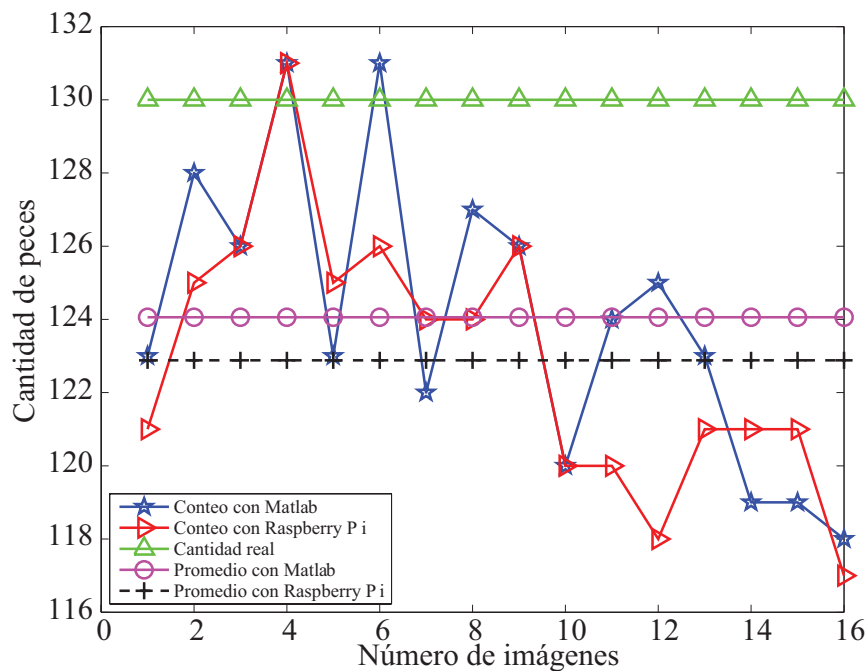


Figura 49: Resultados de contabilizar 130 Mollies (*Poecilia Sphenops*) con una talla promedio de 0.5 cm.

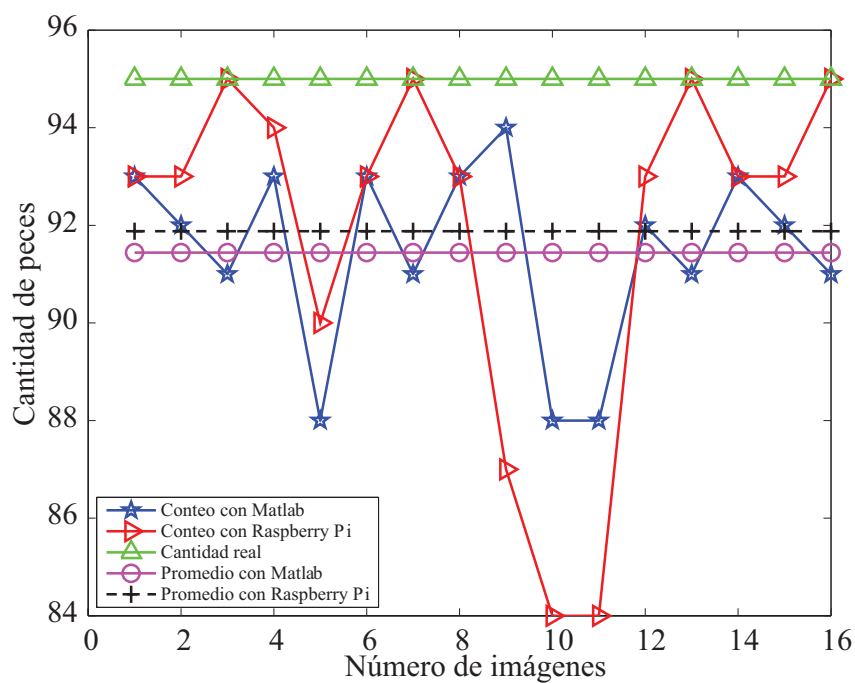


Figura 50: Resultados de contabilizar 95 Mollies (*Poecilia Sphenops*) con una talla promedio de 0.5 cm.

La figura 51 muestra los resultados de contabilizar 50 Mollies, en ella se puede observar como el conteo con la Raspberry Pi usando GNU Octave se encuentra mínimamente por abajo del conteo con la PC usando Matlab, el valor promedio contabilizado con PC es de 48.19, mientras que con Raspberry Pi el valor promedio es de 48.31, el porcentaje de exactitud con PC es de 96.38% mientras que con Raspberry Pi es de 96.63%.

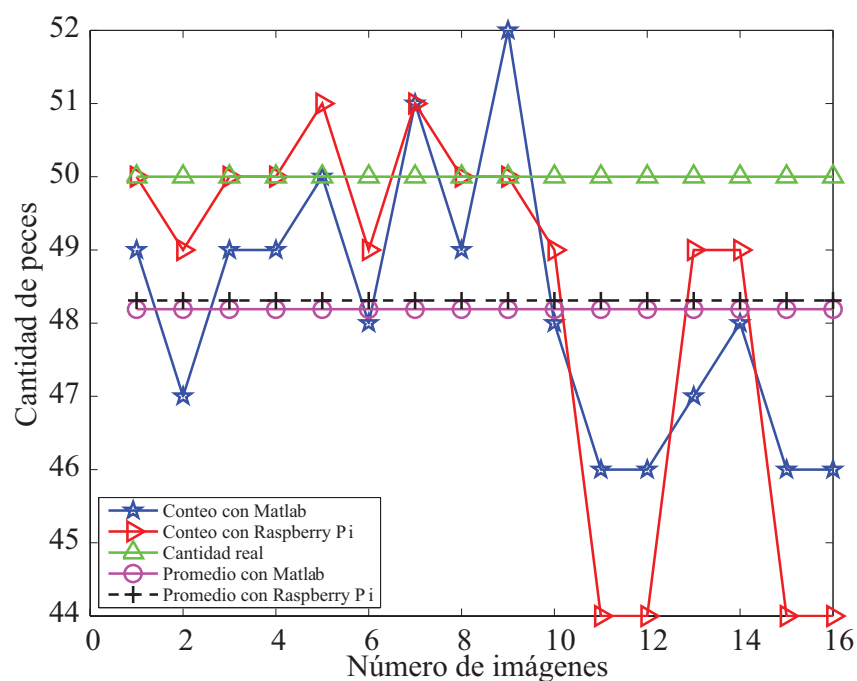


Figura 51: Resultados de contabilizar 50 Mollies (*Poecilia Sphenops*) con una talla promedio de 0.5 cm.

En la tabla XI, se presentan los resultados del conteo de peces nuevamente a partir de 16 imágenes capturadas para diferentes cantidades de Mollies (*Poecilia sphenops*) con una talla promedio de 0.5 cm. y un tiempo de procesamiento del sistema por imagen de 230 segundos. En la primer columna se indica la especie, en la segunda columna se presenta la cantidad real de peces presentes en el acuario, en la tercer columna se indican los peces contabilizados por el sistema propuesto con Raspberry Pi y en la cuarta columna se puede observar la exactitud del sistema. El promedio de exactitud del sistema es de 94.19%.

Tabla XI: Conteo de peces de la especie Mollies (*Poecilia Sphenops*) con tamaño de 0.5 cm usando el sistema contador automático con Raspberry Pi.

<b>Especie</b>	<b>Cantidad real de peces en el acuario</b>	<b>Peces contabilizados por el sistema propuesto</b>	<b>Exactitud del sistema (%)</b>
Mollies ( <i>Poecilia sphenops</i> ) Tamaño de 0.5 cm.	130	122.88	94.52
	120	110.00	91.67
	110	100.00	90.91
	95	91.88	96.71
	80	77.00	96.25
	70	65.00	92.86
	50	48.31	96.38
		<b>Promedio</b>	<b>94.19</b>

Dado que se conoce la cantidad real de peces (contados previamente de forma manual), al ejecutar el algoritmo este genera un resultado sobre el conteo de peces, entonces es posible conocer la exactitud mediante el cálculo de la tasa de exactitud “accuracy rate”(AR) (Fan y Liu, 2013). Es de resaltar que la precisión del algoritmo es variable debido principalmente a tres factores: i) si los peces son menores a 3mm no es posible contabilizarlos, ii) si el pez se encuentra justo bajo la luz blanca reflejada en el agua (saturación) y ninguna parte del pez sobresale de dicha zona, en ese momento para el algoritmo ese pez se ignora, generando un error y iii) cuando un pez se coloca en la orilla del recipiente y al momento de realizar la calibración del recorte del marco

del acuario (figura 22), este pez es borrado y por lo tanto no es contabilizado.

La mayoría de los trabajos reportados en la literatura (Fan y Liu, 2013), (Labuguen *et al.*, 2012), (Zheng y Zhang, 2010), (Zion, 2012), (Zion *et al.*, 2008) y (Zion *et al.*, 2007) usan computadora con el software Matlab y/o herramientas de visión, por lo que se considera conveniente hacer un pequeño análisis sobre el costo que este sistema representa para realizar la función única de contabilizar peces y la ventaja que significaría realizar una migración a un sistema embebido de código abierto, es por esta razón que durante el desarrollo de este trabajo se ha hecho mención del bajo costo del sistema propuesto basado en Raspberry Pi, en la tabla XII se muestra los principales costos en cuanto a hardware, software y cámara de ambos sistemas, por lo que se observa que el sistema basado en Raspberry Pi es más económico, además en cuanto a consumo de energía, la Raspberry Pi consume menos que una PC con Matlab.

Tabla XII: Costo en dólares de los sistemas usados para el conteo de peces, sistema embebido (Raspberry Pi) y PC (MacBook Pro).

	<b>Hardware (Dlls)</b>	<b>Software (Dlls)</b>	<b>Cámara (Dlls)</b>	<b>Costo Total (Dlls)</b>
Sistema Embebido (Raspberry Pi)	\$40.00	\$0.00 (Gratis)	\$15.00	\$55.00
Computadora Personal (PC) (MacBook Pro)	\$900.00	\$180.00	\$60.00	\$1140.00

## IV.4 Comparación de resultados entre ambas plataformas

En la tabla XIII se presentan los resultados de las pruebas de conteo a 3 tallas de peces: i) 350 peces Guppies de 1.2 cm de tamaño promedio, ii) 195 peces Guppies de 2.3 cm de tamaño promedio y iii) 130 peces Mollies de 0.5 cm de tamaño promedio. En la primera columna se presentan las diferentes especies de peces empleadas, en la segunda columna se indica el tamaño promedio de los peces, en la tercera columna se

muestra la cantidad real de peces, la cuarta y quinta columna muestran la desviación estándar obtenida utilizando PC y Raspberry Pi respectivamente, finalmente la sexta y séptima columna muestran el nivel de exactitud del instrumento propuesto, la sexta por PC y la séptima por la Raspberry Pi. El promedio de exactitud usando PC es de 96.19% mientras que usando Raspberry Pi es de 96.64%, lo que significa que el conteo es confiable. Se puede observar que la exactitud del conteo es independiente del tipo de peces y de la talla.

Tabla XIII: Pruebas de conteo de diferentes especies de peces y diferentes tallas obteniendo la desviación estándar y exactitud del instrumento implementado.

Tipo de pez	Tamaño (cm)	Cantidad real de peces	Desviación Estándar		Exactitud del instrumento propuesto (%)	
			PC	Raspberry Pi	PC	Raspberry Pi
Guppies	1.2	350	7.10	5.49	95.67	97.89
Guppies	2.3	195	4.45	3.06	95.92	97.50
Mollies	0.5	130	2.87	2.31	97.00	94.52
			<b>Promedio</b>		<b>96.19</b>	<b>96.64</b>

## IV.5 Comparación de resultados con trabajos relacionados

En la tabla XIV, se presentan los resultados de comparación de exactitud promedio del método propuesto versus los métodos reportados en el estado del arte. Se puede observar que el método con mayor exactitud promedio es el reportado por (Fan y Liu, 2013) quienes utilizan “Least Square Support Vector Machine” (LS-SVM) con un valor de 98.73%. Respecto al método que emplea “Propagation Neural Network” (BPNN) desarrollados por (Newbury *et al.*, 1995) y (Fan y Liu, 2013) obtienen 94.00% y 94.41% respectivamente. El método propuesto alcanza exactitud promedio de 96.19% implementado en una PC con Matlab y en el sistema embebido propuesto un valor promedio de 96.64%, por lo tanto, la exactitud del método propuesto se encuentra

ligeramente superior a BPNN, siendo una diferencia de 2.23% con respecto a Raspberry Pi. La exactitud promedio en nuestro método es 2.1% menor que la exactitud promedio del método LS-SVM, sin embargo la densidad de peces en nuestras pruebas es 3 veces superior, ya que sus pruebas incluyen hasta 100 peces y las nuestras hasta 350 peces con un volumen de agua ligeramente menor. De acuerdo a (Fan y Liu, 2013), el error de conteo en el método LS-SVM se incrementa en proporción a la densidad de peces. Por lo tanto, la técnica propuesta es competitiva con los métodos de clasificación del estado del arte.

Tabla XIV: Comparación de resultados con los métodos de clasificación del estado del arte.

	Método propuesto (PC)	Método propuesto (Raspi)	Exactitud LS-SVM (Fan, 2013)	Exactitud BPNN (Fan, 2013)	Exactitud BPNN (Newbury, 1995)
Exactitud Promedio (%)	96.19	96.64	98.73	94.41	94.00

# Capítulo V

## Conclusiones generales y aportaciones

### V.1 Conclusiones

En este trabajo de tesis doctoral se aborda la forma de solucionar un problema real respecto a la contabilización automática de peces. Se inicia con la generación de plantillas de los peces a identificar y basándose en los momentos invariantes de  $H_u$ , se encuentra que es posible identificar el 100% de los peces con la limitante que no son imágenes de peces reales.

Posterior al uso de los momentos invariantes de  $H_u$ , se empieza a trabajar con peces vivos del tipo ornamental dentro de un acuario, a través de un método basado en procesamiento digital de imágenes implementado en una PC con Matlab. El conteo resulta sumamente eficiente y muy rápido (del orden de los milisegundos). El problema encontrado en este sistema es la portabilidad y su alto costo, dado que se necesita tener físicamente una computadora personal con matlab como parte del sistema contador de peces, lo cual resulta costoso para el procesado de la información.

Finalmente el algoritmo propuesto se implementa en el sistema embebido Raspberry Pi 2, por lo que se migra todo el código fuente usado en Matlab dentro de la PC hacia el software GNU Octave, el cual es de código abierto y distribución gratuita. Los resultados son muy aceptables y muy similares a los obtenidos con PC, con la única limitante del tiempo de procesamiento, siendo del orden de 230 segundos con Raspberry Pi. El sistema final resulta sumamente atractivo para granjas de peces ornamentales.

De los resultados experimentales obtenidos, se valida la funcionalidad del instrumento operando en rangos de confiabilidad aceptables. Su diseño está pensado para integrarse en cualquier etapa de un proceso piscícola, aún cuando se vislumbra su amplia potencialidad en aplicaciones acuícolas y de comercialización de diversas especies marinas. La instrumentación y elementos que integran al sistema contador son de bajo costo, es de fácil manipulación y escalable tecnológicamente, lo que lo hace ideal para que los acuicultores lleven acabo buenas prácticas en sus procesos de cultivo.

## V.2 Aportaciones

- Estudio empírico sobre el color de fondo óptimo del acuario, obteniendo el color amarillo con un error sistemático menor.
- Implementación e integración de un conjunto de algoritmos que utilizan técnicas de procesamiento digital de imágenes en un sistema embebido, para el conteo de peces en tiempo real.
- Desarrollo del sistema contador de peces en tiempo real empleando un dispositivo embebido de alto rendimiento computacional, el cual puede ser útil para actividades de investigación y en el sector productivo para granjas piscicultoras.
- Este software puede ser una herramienta de apoyo en los cursos de procesamiento digital de imágenes que se imparten en posgrado y licenciatura dentro de la UABC, adicionalmente se fortalece la didáctica de las técnicas de procesamiento digital de imágenes y se valida en la resolución de un problema real.

## V.3 Trabajos a futuro

- El sistema solo se probó con peces ornamentales, por lo que sería deseable realizar experimentos en diferentes especies marinas, tales como moluscos, crustáceos, así como en diferentes peces de agua dulce y salada.
- En lo referente al lenguaje de programación, sería deseable migrar el código desarrollado en GNU Octave a openCV, esperando con esto que se reduzcan los tiempos de procesamiento.
- Paralelizar los algoritmos propuestos para mejorar el tiempo de procesamiento.
- Probar los algoritmos propuestos en otros sistemas embebidos de alto rendimiento computacional.
- Probar el algoritmo empleando otras fuentes de luz con diferentes longitudes de onda, es decir con otros colores.
- El sistema propuesto, es factible de implementarse en ambientes reales de granjas piscicultoras principalmente de dos maneras: i) colocando la cámara digital del contador de peces sobre un ducto que une dos grandes estanques, donde uno de los estanques contiene el total de peces a contabilizar y mediante una recirculación forzada se harán pasar los peces al otro estanque, en el proceso el sistema propuesto obtendrá imágenes de los peces cada cierto tiempo, de tal forma que en tiempo real se contabilizarán los peces que van cruzando, cabe mencionar

que el tiempo de captura de imágenes dependerá de la velocidad del flujo de agua y ii) mediante un arreglo matricial de sistemas embebidos propuestos colocados estratégicamente para cubrir la totalidad del área de un gran estanque acuícola, es decir, cada una de las cámaras dará cobertura a una sub-área distribuida equitativamente, de tal forma que cada sistema funcionará como un contador independiente y enviará el resultado de su conteo a un sistema coordinador central, el cual realizará el conteo total. Para este caso, se propone interconectar los sistemas mediante una topología de red tipo estrella con comunicación Ethernet o WiFi, las cuales vienen integradas en el sistema embebido propuesto.

# Bibliografía

- Agrawal, N. y Singhal, S. (2015). Smart drip irrigation system using Raspberry Pi and Arduino. En *Computing, Communication and Automation (ICCCA), 2015 International Conference*, páginas 928–932. IEEE.
- Akitomi Minoru, Kodaira Takayuki, T. M. (2016). Fish counter, fish counting method, prediction device for number of fish, prediction method for number of fish, fish counter system and prediction system for number of fish. *Espacenet*, **2016**(JP20140208835 20141010).
- Aksentijević, S., Krnjak, D., y Tijan, E. (2015). Logistics environment awareness system prototype based on modular internet of things platform. *Pomorstvo: Scientific Journal of Maritime Research*, **29**(2): 170–179.
- Amir, S., Siddiqui, A. A., Ahmed, N., y Chowdhry, B. S. (2014). Implementation of line tracking algorithm using Raspberry Pi in marine environment. En *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, páginas 1337–1341. IEEE.
- Arey Clyde D., A. R. D. y. T. W. B. (1977). Apparatus for measuring, weighing and counting fish. *United States Patent*, **1977**(US5637838).
- Baker, Spilde, B. y Gruner (2007). Ultrasonic measurements in fish-water pipe flows. En *5th World Congress on Industrial Process Tomography*, páginas 456–463.
- Castillo, M. R. (2012). Procesamiento digital de imágenes empleando filtros espaciales. *Tesis de licenciatura, Universidad Autónoma de Baja California*.
- Catarino, Aguilar O, P. M. J. A. y. F. F. A. A. (2015). Sistema de monitoreo inalámbrico multi-punto mediante plataforma Arduino y Raspberry-Pi. En Electro, editor, *Congreso Internacional en Ingeniería Electrónica*, Vol. 37, páginas 455–460. Electro.

- Cristian Morelos Hernández, Aldrin Barreto Flores, V. E. B. L. (2015). Sistema de seguridad basado en Internet de las cosas y electric imp. En Electro, editor, *Congreso Internacional en Ingeniería Electrónica.*, Vol. 37, páginas 460–465. Electro.
- David, D. P. (2013). Fish counter. *United States Patent*, **2013**(US00D687731S).
- Duszynski, G. J. (2011). Portable container mounted counter for fishing, hunting, and other outdoor activities. *United States Patent*, **2011**(US08044337).
- Fabic, J., Turla, I., Capacillo, J., David, L., y Naval, P. (2013). Fish population estimation and species classification from underwater video sequences using blob counting and shape analysis. En *Underwater Technology Symposium (UT), 2013 IEEE International*, páginas 1–6. IEEE.
- Fan, L. y Liu, Y. (2013). Automate fry counting using computer vision and multi-class least squares support vector machine. *Aquaculture*, **380**: 91–98.
- Ferrero, F., Campo, J., Valledor, M., y Hernando, M. (2014). Optical systems for the detection and recognition of fish in rivers. En *Systems, Signals and Devices (SSD), 2014 11th International Multi-Conference on*, páginas 1–5. IEEE.
- Fix, E. y Hodges Jr, J. L. (1951). Discriminatory analysis-nonparametric discrimination: consistency properties. Reporte técnico, DTIC Document.
- Fix, E. y Hodges Jr, J. L. (1952). Discriminatory analysis-nonparametric discrimination: Small sample performance. Reporte técnico, DTIC Document.
- Fox, C. R. (1994). Fish-bite indicator. *United States Patent*, **1994**(US5321391).
- Frenzel, L. (2015). *Principles of electronic communication systems*. McGraw-Hill.
- Gonzalez, R. C. y Woods, R. E. (2016). Digital image processing. *Pearson*.
- Hernández-Ontiveros J. M., García-Guerrero E. E., I.-G. E. y L.-B. O. R. (2014). Implementation of an algorithm for fish counting and identification based on

- Hu moments for aquaculture applications. *Congreso Internacional en Ciencias Computacionales*.
- Hiroshi, K. (2007). Sistema de procesamiento de imagen y método de procesamiento de imagen. *Patentes de España*, **2007**(ES2280096).
- Hu, M.-K. (1962). Visual pattern recognition by moment invariants. *IRE transactions on information theory*, **8**(2): 179–187.
- Huang, D.-Y. y Wang, C.-H. (2009). Optimal multi-level thresholding using a two-stage otsu optimization approach. *Pattern Recognition Letters*, **30**(3): 275–284.
- Ibrahim, D. (2014). Using the Raspberry Pi computer in automation applications. *Electronics World*, **120**(1941): 22–26.
- Islam, M. M., Azad, M. S. U., Alam, M. A., y Hassan, N. (2014). Raspberry Pi and image processing based electronic voting machine (EVM). *Int J Sci Eng Res*, **5**(1).
- Karplus, I., Alchanatis, V., y Zion, B. (2005). Guidance of groups of Guppies (*Poecilia reticulata*) to allow sorting by computer vision. *Aquacultural engineering*, **32**(3): 509–520.
- Kenney John, F. (1939). *Mathematics Of Statistics*. D. Van Nostrand Company, Inc; Toronto; New York; London.
- Labuguen, R., Volante, E., Causo, A., Bayot, R., Peren, G., Macaraig, R., Libatique, N., y Tangonan, G. (2012). Automated fish fry counting and schooling behavior analysis using computer vision. En *Signal Processing and its Applications (CSPA), 2012 IEEE 8th International Colloquium on*, páginas 255–260. IEEE.
- Lamine, H. y Abid, H. (2014). Remote control of a domestic equipment from an Android application based on Raspberry Pi card. En *Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2014 15th International Conference on*, páginas 903–908. IEEE.

- Latham, M. W. (1993). Side-looking fish finder. *United States Patent*, **1993**(US5260912).
- Leccese, F., Cagnetti, M., y Trinca, D. (2014). A smart city application: A fully controlled street lighting isle based on Raspberry-Pi card, a ZigBee sensor network and WiMAX. *Sensors*, **14**(12): 24408–24424.
- Li, X., Hao, J., Qin, H., y Chen, L. (2015). Real-time fish localization with binarized normed gradients. En *Oceans 2015-MTS/IEEE Washington*, páginas 1–5. IEEE.
- Lofaso, D. P. (2004). Electronic fish counter apparatus. *United States Patent*, **2004**(US6809658).
- Loh, B. C., Raman, V., y Then, P. H. (2011). First prototype of aquatic tool kit: Towards low-cost intelligent larval fish counting in hatcheries. En *Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on*, páginas 192–195. IEEE.
- Maki Kazunari, T. Y. (2007). Fish counting method, fish photographing device, and fish counting system. *Patente de Japón*.
- Martínez, S. E. (2015). *Sistema de visión por computadora para contabilizar larvas de peces*. Tesis de maestría, Universidad Autónoma de Baja California.
- Matcham, D. A. (1990). Fish counter. *United States Patent*, **1990**(D310046).
- Menin, A. S. (1984). Salmon counter with separate king salmon tabulator. *United States Patent*, **1984**(US4439844).
- Menin, A. S. y B. R. M. (1977). Fish counter. *United States Patent*, **1977**(US004034331).
- Menin, A. S. y F. M. E. (1980). Fish counter with scanning transducer. *United States Patent*, **1980**(US4225951).

- Mondaca, S. C. E. (2017). *Sistema electrónico para la contabilización de peces*. Tesis de maestría, Universidad Autónoma de Baja California.
- Moore, H. M. (2007). *MATLAB para ingenieros*. Pearson.
- Morais, E. F., Campos, M. F. M., Padua, F. L., y Carceroni, R. L. (2005). Particle filter-based predictive tracking for robust fish counting. En *XVIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'05)*, páginas 367–374. IEEE.
- Moyle, P. B. C. y Joseph, J. (2004). *Fishes: an introduction to ichthyology*. Número 597 MOY en 4. Prentice-Hall.
- Navarro, R. A. (2015). *Instrumentación electrónica para aplicaciones en acuicultura*. Tesis de maestría, Universidad Autónoma de Baja California.
- Nelson, J. S., Grande, T. C., y Wilson, M. V. (2016). *Fishes of the World*. John Wiley and Sons.
- Newbury, P. F., Culverhouse, P. F., y Pilgrim, D. A. (1995). Automatic fish population counting by artificial neural network. *Aquaculture*, **133**(1): 45–55.
- Nguyen, H.-Q., Loan, T. T. K., Mao, B. D., y Huh, E.-N. (2015). Low cost real-time system monitoring using Raspberry Pi. En *2015 Seventh International Conference on Ubiquitous and Future Networks*, páginas 857–859. IEEE.
- Nikhade, S. G. (2015). Wireless sensor network system using Raspberry Pi and Zigbee for environmental monitoring applications. En *Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), 2015 International Conference on*, páginas 376–381. IEEE.
- Otsu, N. (1975). A threshold selection method from gray-level histograms. *Automatica*, **11**(285-296): 23–27.
- Pérez López, C. (2002). *Matlab y sus aplicaciones en las ciencias y la ingeniería*. Editorial Pearson Educación, SA, Madrid.

- Reyes Ibarra Luis Alberto, Herrera Castillo Perla Alejandra, R. C. C. (2015). Controlador de riego para un invernadero empleando el Internet de las cosas y Raspberry Pi B+. En Electro, editor, *Congreso Internacional en Ingeniería Electrónica.*, Vol. 37, páginas 343–348. Electro.
- Richardson, D. y S. A. (2006). Combined storage bin and fish counter. *United States Patent*, **2006**(D532860).
- Sapes, J. y Solsona, F. (2016). Fingerscanner: Embedding a fingerprint scanner in a Raspberry Pi. *Sensors*, **16**(2): 220.
- Sarango Chuquimarca, F. E. (2014). *Diseño e implementación del prototipo de un sistema domótico para la medición del consumo de agua potable a través de Internet y correo electrónico*. Tesis de maestría, Universidad Nacional de Loja.
- Schumann-Bölsche, D. y Schön, A.-M. (2015). A Raspberry in Sub-Saharan Africa? chances and challenges of Raspberry Pi and Sensor Networking in Humanitarian Logistics. *Procedia Engineering*, **107**: 263–272.
- Shen, W., Guo, X., Wu, C., y Wu, D. (2011). Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm. *Knowledge-Based Systems*, **24**(3): 378–385.
- Song, R., Chen, X., Shen, C., y Zhang, H. (2014). Modeling fog drift using back-propagation neural network optimized by artificial fish swarm algorithm. *Journal of Sensors*, **2014**.
- Takagi Yoichi, Kato Masayasu, M. T. (1997). Method and apparatus for counting underwater objects using an ultrasonic wave. *United States Patent*, **1997**(US5692064).
- Toh, Y., Ng, T., y Liew, B. (2009). Automated fish counting using image processing. En *Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on*, páginas 1–5. IEEE.

- Villarejo, J. I. D. (2002). *Atlas y libro rojo de los peces continentales de España*. Museo Nacional de Ciencias Naturales, Dirección General de Conservación de la Naturaleza.
- Watson, G., Mason, A., y Ackroyd, R. (2014). *Social Engineering Penetration Testing: Executing Social Engineering Pen Tests, Assessments and Defense*. Syngress.
- Zheng, X. y Zhang, Y. (2010). A fish population counting method using fuzzy artificial neural network. En *Progress in Informatics and Computing (PIC), 2010 IEEE International Conference on*, Vol. 1, páginas 225–228. IEEE.
- Zion, B. (2012). The use of computer vision technologies in aquaculture—a review. *Computers and electronics in agriculture*, **88**: 125–132.
- Zion, B., Shklyar, A., y Karplus, I. (1999). Sorting fish by computer vision. *Computers and Electronics in Agriculture*, **23**(3): 175–187.
- Zion, B., Shklyar, A., y Karplus, I. (2000). In-vivo fish sorting by computer vision. *Aquacultural Engineering*, **22**(3): 165–179.
- Zion, B., Alchanatis, V., Ostrovsky, V., Barki, A., y Karplus, I. (2007). Real-time underwater sorting of edible fish species. *Computers and Electronics in Agriculture*, **56**(1): 34–45.
- Zion, B., Alchanatis, V., Ostrovsky, V., Barki, A., y Karplus, I. (2008). Classification of Guppies' (*Poecilia reticulata*) gender by computer vision. *Aquacultural engineering*, **38**(2): 97–104.

## Apéndice A

# Dibujo y Perspectiva de la Estructura Mecánica

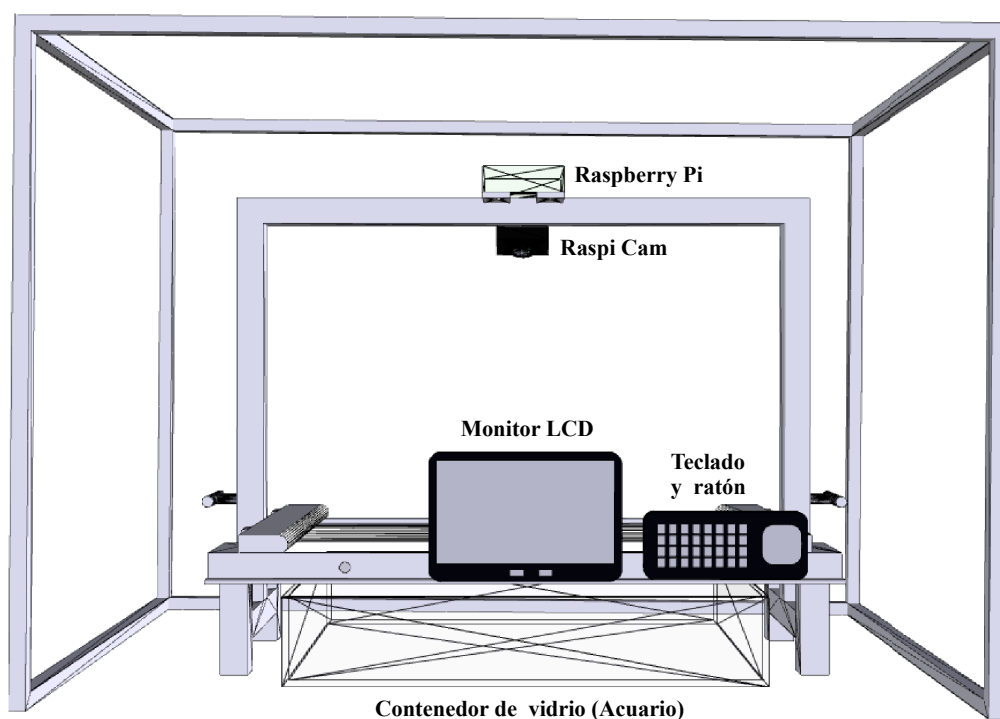


Figura 52: Dibujo del contador de peces.

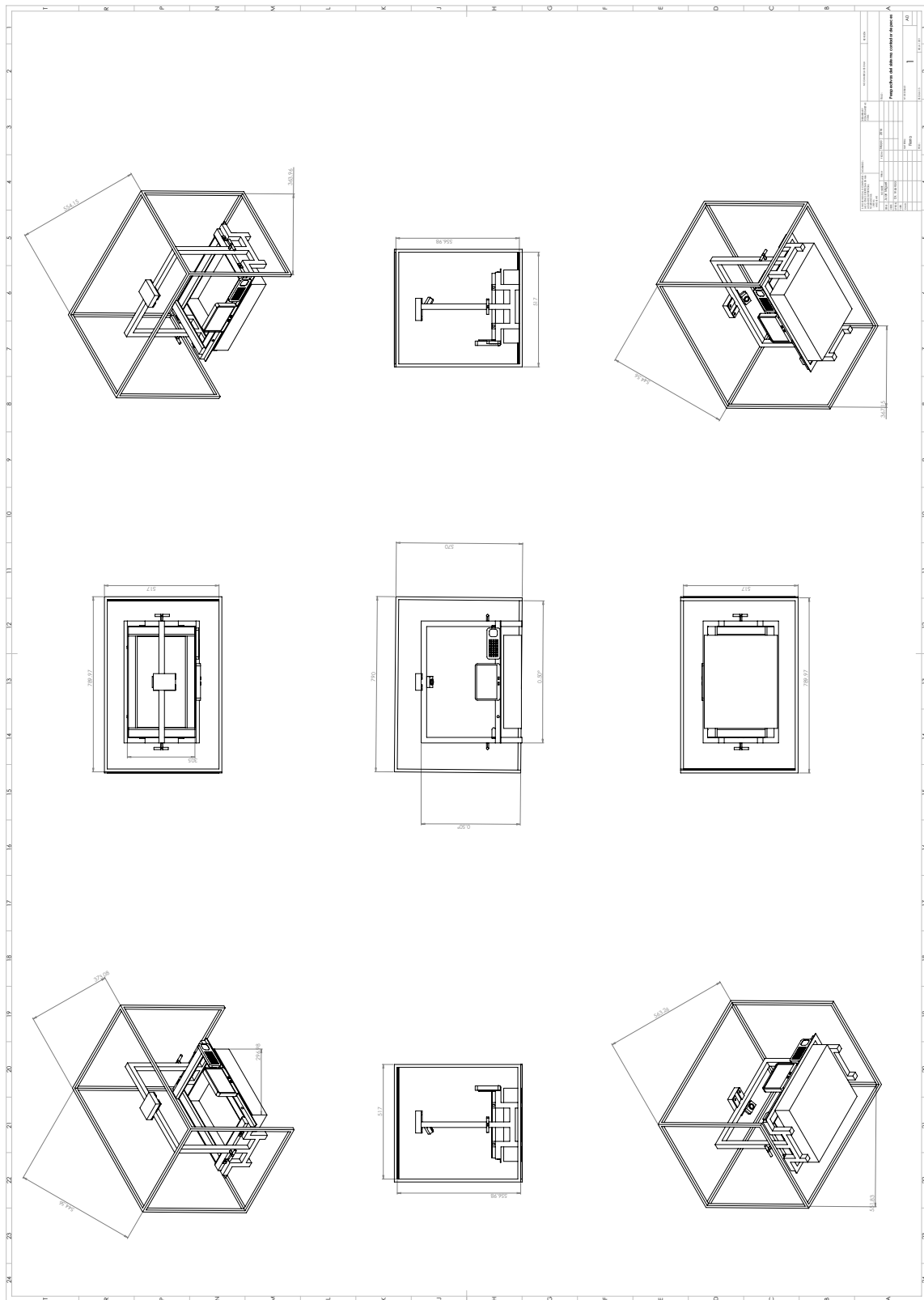


Figura 53: Perspectivas del contador de peces.

# Apéndice B

## Producción Académica

**Hernández-Ontiveros Juan Miguel**, Inzunza-González Everardo, García-Guerrero Enrique Efren y López-Bonilla Oscar Roberto. (2017). Aparato y método contador automático de peces usando procesamiento digital de imágenes. *Patente en trámite*.

**Hernández-Ontiveros Juan Miguel**, Inzunza-González Everardo, García-Guerrero Enrique Efren, López-Bonilla Oscar Roberto, Infante-Prieto Sergio Omar, Cárdenas-Valdez José Ricardo y Tlelo-Cuautle E. (2017). Development and implementation of a fish counter by using an embedded system. Sometido en: *Computers and Electronics in Agriculture*.

**Hernández-Ontiveros Juan Miguel**, Inzunza-González Everardo, García-Guerrero Enrique Efren, López-Bonilla Oscar Roberto y Esqueda Elizondo José Jaime. (2016). Sistema domótico controlado por correo electrónico. Congreso Internacional de Ingeniería Electrónica.

**Hernández-Ontiveros Juan Miguel**, Mondaca-Santillan César Eligio, García-Guerrero Enrique Efren, Inzunza-González Everardo y López-Bonilla Oscar Roberto. (2015). Sistema domótico controlado por Gmail. XXII Jornadas de Ingeniería Arquitectura y Diseño, FIAD-UABC.

**Hernández-Ontiveros Juan Miguel**, García-Guerrero Enrique Efren, Inzunza-González Everardo y López-Bonilla Oscar Roberto. (2014). Implementation of an algorithm for fish counting and identification based on Hu moments for aquaculture applications. Congreso Internacional en Ciencias Computacionales.

**Hernández-Ontiveros Juan Miguel**, García-Guerrero Enrique Efren, Inzunza-González Everardo y López-Bonilla Oscar Roberto. (2014). Sistema contador de especies marinas mediante el procesamiento de imágenes digitales. Reporte técnico entregado a Granja Brekarand, Especialistas en Peces Ornamentales.

# Apéndice C

## Glosario de términos

°C. Grados Centígrados.

%. Porcentaje.

AR (Accuracy Rate). Tasa de Exactitud.

ARM (Advanced RISC Machine). Máquina RISC Avanzada.

BPNN (Back Propagation Neural Network). Redes Neuronales de Retro-Propagación.

CCD (Charge-Coupled Device). Dispositivo de Carga Acoplada.

cm. Centímetros.

CMOS (Complementary Metal-Oxide-Semiconductor). Semiconductor Complementario de Óxido Metálico.

CPU (Central Processing Unit). Unidad Central de Procesamiento.

CSi (Camera Serial interface) Interfaz serie para cámara.

DDR3 (Double Data Rate). Doble Velocidad de Transmisión de Datos.

GB (GigaByte). GigaByte.

GHz. GigaHertz.

GNU (GNU's Not Unix). Acrónimo recursivo para GNU No Unix.

GNU Octave. Programa libre para realizar cálculos numéricos.

GPU (Graphics Processing Unit). Unidad de procesamiento Gráfico.

GSM (Global System for Mobile Communications). Sistema Global para Comunicaciones Móviles.

HDMI (High-Definition Multimedia Interface). Interfaz Multimedia de Alta Definición

IoT (Internet of Things). Internet de las cosas.

LabVIEW<sup>TM</sup> (Laboratory Virtual Instrument Engineering Workbench). Plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico.

LCD (Liquid Crystal Display). Pantalla de Cristal Líquido.

LED (Light-Emitting Diode). Diodo Emisor de Luz.

Linux. Sistema operativo de software libre.

LS-SVM (Least Square Support Vector Machine). Máquina de Vectores de Soporte y Mínimos Cuadrados.

Matlab (MATrix LABoratory). Laboratorio de Matrices, es una herramienta de software matemático.

MB/s. MegaByte por segundo.

MHz. MegaHertz.

MicroSD (Secure Digital). Dispositivo en formato de tarjeta de memoria para dispositivos portátiles.

microSDHC (Secure Digital High Capacity). Formato de tarjeta de memoria flash.

mm. Milímetros

MP (MegaPixel). Megapíxeles.

ms. Milisegundos.

openCV. Biblioteca libre de visión artificial originalmente desarrollada por Intel.

PC (Personal Computer). Computadora personal.

pH. Medida de acidez o alcalinidad de una disolución. Potencial de Hidrógeno.

RAM (Random Access Memory). Memoria de Acceso Aleatorio.

RGB (Red, Green, Blue). Modelo de color basado en los colores primarios Rojo, Verde y Azul.

SDRAM (Synchronous Dynamic Random-Access Memory). Familia de memorias dinámicas de acceso aleatorio.

SMS (Short Message Service). Servicio de mensajes cortos.

SoC (System on a Chip). Sistema en un chip.

USB (Universal Serial Bus). Bus universal en serie.

$V_{ac}$  (Voltage with Alternating current). Voltaje con corriente alterna.

W (Watt). Medida para representar la potencia eléctrica.

Web. Red informática.

WiMAX (Worldwide Interoperability for Microwave Access). Interoperabilidad para el

acceso a microondas.

ZigBee. Estándar de comunicaciones inalámbricas IEEE 802.15.4.