

# Metodología de desarrollo de sistemas de detección y seguimiento de peces mediante tecnología láser y visión artificial con inteligencia artificial

Autor: Ángel José Rico Díaz

---

Tesis doctoral UDC / 2022

Director/a: Juan Ramón Rabuñal Dopico

Codirector/a: Jerónimo Puertas Agudo

Programa de doctorado en Tecnologías de la Información y las Comunicaciones



Agradecer sinceramente a todos los que han colaborado en esta tesis de una u otra manera, especialmente a Álvaro, Dani, Esteban, Miki, Gonzalo, Omar, Adrián, Alberto y Marcos.

Ángel Rico

# Agradecimientos

Esta tesis no habría sido posible desarrollarla sin el apoyo, tanto logístico como de personal, del grupo de investigación RNASA-Imedir, del CITIC (Centro de Investigación en Tecnologías de la Información y las Comunicaciones de A Coruña) y del CITEEC (Centro de Innovación en Construcción e Ingeniería Civil). También se agradece la cesión de las instalaciones del Aquarium Finisterrae de A Coruña, en donde se realizaron algunos ensayos.

Este trabajo ha sido cofinanciado con fondos FEDER y por el Ministerio Español de Economía y Competitividad. Subprograma estatal de formación del Programa Estatal de Promoción de Talento y su Empleabilidad en I+D, en el marco del Plan Estatal de Investigación Científica y técnica y de Innovación 2013-2016 (FPI Convocatoria 2013) (Ref. del proyecto CGL2012-34688 Ref. de la ayuda BES-2013-063444).



# Resumen

En esta tesis se presenta una metodología de detección, seguimiento y medición de peces bajo el agua utilizando técnicas de teledetección. En la mayoría de las aplicaciones en las que se precisa la detección del pez y la medida de su tamaño, es imprescindible que se encuentren en su hábitat nadando libremente y sin ningún dispositivo eléctrico o mecánico colocado en el propio pez. De esta manera no se condiciona su comportamiento ni se somete a estrés que pudiera desencadenar situaciones no deseadas. Las técnicas de teledetección basadas en luz visible e infrarroja permiten detectar el pez sin contacto físico, sin embargo, es necesario adecuarlas para su correcto funcionamiento bajo el agua. En esta tesis se presenta esta metodología utilizando visión artificial y tres aplicaciones prácticas con dispositivos de detección basados en principios físicos diferentes. También se pretende indagar en la viabilidad de utilizar dispositivos, inicialmente diseñados para funcionar en entornos fuera del agua, que pueden ser adaptados mediante algoritmos informáticos en el seguimiento de objetos bajo el agua, y principalmente en la detección y medición de peces en su entorno acuático.

# Resumo

Nesta tese preséntase unha metodoloxía de detección, seguimento e medición de peixes baixo a auga utilizando técnicas de teledetección. Na maioría das aplicacións nas que se precisa a detección do peixe e a medida do seu tamaño, é imprescindible que se atopen no seu hábitat nadando libremente e sen ningún dispositivo eléctrico ou mecánico colocado no propio peixe. Deste xeito non se condiciona o seu comportamento nin se somete a estrés que puidese desencadear situacións non deseadas. As técnicas de teledetección baseadas na luz visible e infrarvermella permiten detectar o peixe sen contacto físico, con todo, é preciso adecualas para o seu correcto funcionamento baixo a auga. Nesta tese preséntase esta metodoloxía utilizando visión artificial e tres aplicacións prácticas con dispositivos de detección baseados en principios físicos diferentes. Tamén se pretende indagar na viabilidade de utilizar dispositivos inicialmente deseñados para funcionar en contornas fora da auga, que poden ser adaptados mediante algoritmos informáticos no seguimento de obxectos baixo a auga e, principalmente, na detección e medición de peixes na súa contorna acuática.

# Abstract

This PhD thesis presents a methodology to underwater fish detection, tracking and measuring, using remote sensing techniques. In most cases the applications where fish detection and size measurement is required, it is essential that they are in their hábitat, swimming freely and without any electrical or mechanical device attached to the fish itself. In this way, their behavior is not conditioned or they are subjected to stress that could cause unwanted situations. Remote sensing techniques based on visible light and infrared light allow the fish to be detected without physical contact however, it is necessary to adapt them for their correct operation under water. In this thesis, this methodology using computer vision and three practical applications with detection devices base don different physical principles are presented. In addition, this work studies the feasibility of using devices, initially designed to work in outside the water enviroments, which can be adapted by means of computer algorithms in the tracking of underwater objects, and mainly in the detection and measurement of fish in their environment.

# Índice

<b>1. RESUMEN LARGO .....</b>	<b>14</b>
1.1. CASTELLANO .....	14
1.2. GALEGO .....	15
1.3. ENGLISH .....	17
<b>2. INTRODUCCIÓN.....</b>	<b>19</b>
2.1. ESTRUCTURA DEL DOCUMENTO .....	21
<b>3. ESTADO DEL ARTE .....</b>	<b>22</b>
<b>4. METODOLOGÍA .....</b>	<b>34</b>
<b>5. CASO PRÁCTICO 1: UNA APLICACIÓN DE DETECCIÓN DE PECES BASADA EN LA BÚSQUEDA DEL OJO DEL PEZ CON VISIÓN ARTIFICIAL Y REDES DE NEURONAS ARTIFICIALES .....</b>	<b>38</b>
5.1. INTRODUCCIÓN .....	38
5.2. METODOLOGÍA .....	40
1. <i>Dispositivo hardware de captura</i> .....	40
2. <i>Calibración</i> .....	43
3. <i>Transformación de la información capturada</i> .....	45
4. <i>Pre-procesado</i> .....	46
5. <i>Recepción de información en tiempo real</i> .....	50
6. <i>Detección de objetos</i> .....	50
7. <i>Seguimiento de objetos</i> .....	60
8. <i>Obtención del tamaño del pez</i> .....	61
9. <i>Visualización en el ordenador</i> .....	65
5.3. RESULTADOS EXPERIMENTALES .....	65
5.4. CONCLUSIONES .....	69
<b>6. CASO PRÁCTICO 2: DETECCIÓN DE PECES EN ESCALA DE HENDIDURA VERTICAL COMBINANDO TÉCNICAS DE VISIÓN ARTIFICIAL CON LA TECNOLOGÍA LÁSER.....</b>	<b>71</b>
6.1. INTRODUCCIÓN .....	71

6.2.	METODOLOGÍA .....	73
1.	<i>Dispositivos hardware de captura</i> .....	73
2.	<i>Calibración: Validación inicial del uso de láser en agua</i> .....	76
3.	<i>Transformación de la información capturada</i> .....	79
4.	<i>Pre-procesado</i> .....	80
5.	<i>Recepción de la información en tiempo real</i> .....	80
6.	<i>Detección de objetos</i> .....	81
7.	<i>Seguimiento de objetos</i> .....	85
8.	<i>Obtención de mediciones del objeto</i> .....	86
9.	<i>Visualización en tiempo real</i> .....	87
6.3.	RESULTADOS EXPERIMENTALES.....	92
6.4.	CONCLUSIONES .....	95
<b>7.</b>	<b>CASO PRÁCTICO 3: DETECCIÓN DE PECES MEDIANTE LUZ</b>	
	<b>ESTRUCTURADA.....</b>	<b>97</b>
7.1.	INTRODUCCIÓN .....	97
7.2.	METODOLOGÍA .....	97
1.	<i>Dispositivo hardware de captura</i> .....	97
2.	<i>Calibración</i> .....	102
3.	<i>Transformación de la información capturada</i> .....	104
4.	<i>Pre-procesado</i> .....	104
5.	<i>Recepción de la información en tiempo real</i> .....	116
6.	<i>Detección de objetos</i> .....	117
7.	<i>Seguimiento de objetos</i> .....	122
8.	<i>Obtención del tamaño del pez</i> .....	123
9.	<i>Visualización en tiempo real</i> .....	127
7.3.	RESULTADOS EXPERIMENTALES.....	129
7.4.	CONCLUSIONES .....	135
<b>8.</b>	<b>CONCLUSIONES Y TRABAJO FUTURO .....</b>	<b>136</b>
8.1.	CASTELLANO .....	136
8.2.	GALEGO .....	138
8.3.	ENGLISH .....	139
<b>9.</b>	<b>PUBLICACIONES .....</b>	<b>142</b>

<i>Revistas internacionales con impacto</i> .....	142
<i>Revistas internacionales con revisión por pares y sin impacto</i> .....	143
<i>Congresos internacionales</i> .....	146
<i>Capítulos de libros</i> .....	147
<i>Patentes</i> .....	149
<i>Otras publicaciones</i> .....	150
<b>BIBLIOGRAFÍA</b> .....	<b>152</b>

# Índice de figuras

<b>Figura 1:</b> Estructura de una escala de hendidura vertical, ubicada en el Centro de Estudios y Experimentación de Obras Públicas (CEDEX) del Centro de Estudios Hidrográficos (CEH) .....	22
<b>Figura 2 :</b> Estructura del dispositivo Riverwatcher Vaki [42] .....	27
<b>Figura 3:</b> Cámara con leds utilizado por Vaki en el Riverwatcher [42].....	27
<b>Figura 4 :</b> Uno de los contadores de AquaScan, en este caso el CSW27600 [43] .....	28
<b>Figura 5:</b> Contador de peces de <i>Calitri Technology</i> y un posible montaje [44].....	28
<b>Figura 6:</b> Dos modelos de contadores de la empresa Faivre [45] .....	29
<b>Figura 7:</b> Contador de peces de Impex Agency ApS [46].....	29
<b>Figura 8:</b> Unidad del contador de peces SR-1601 de Smith-Root [47].....	30
<b>Figura 9:</b> Muestra los tipos de radiación relacionados con sus longitudes de onda y el espectro visible .....	35
<b>Figura 10:</b> Pasos del algoritmo propuesto y ejemplos de cada paso .....	39
<b>Figura 11:</b> GoPro Hero3+ Black (a) sin carcasa (b) con carcasa estanca .....	42
<b>Figura 12:</b> (a) carcasa estanca para visión estéreo (b) elemento de sincronización de las cámaras (en naranja).....	42
<b>Figura 13:</b> Geometría epipolar.....	43
<b>Figura 14:</b> Proceso de calibración (a) con la plantilla de puntos coplanarios (b).....	45
<b>Figura 15:</b> Mean Shift Analysis .....	47
<b>Figura 16:</b> Ejemplo de imagen filtrada (a) imagen Input (b) imagen Output .....	48
<b>Figura 17:</b> Contornos exteriores e interiores en una imagen.....	50
<b>Figura 18:</b> Cada punto del espacio geométrico (izquierda) es el centro de un círculo en el espacio paramétrico (derecha). Los círculos de dicho espacio se intersecan en (a, b) que corresponde al centro (x, y) en el espacio geométrico. ....	52
<b>Figura 19:</b> características utilizadas: la esclerótica y la pupila del ojo de pez. ....	53
<b>Figura 20:</b> Líneas que delimitan la esclerótica y la pupila del candidato a ojo.....	54
<b>Figura 21:</b> Ejemplo de un ojo de pez (y zoom) correctamente emparejados que está resaltado en azul. El ROI en el que se ubicó el pez está marcado con un rectángulo verde. ....	56
<b>Figura 22:</b> Extracto del conjunto de datos utilizado para el entrenamiento y la validación de RNA.....	59

<b>Figura 23:</b> (a) Imagen real (b) mapa de disparidad de la imagen.....	61
<b>Figura 24:</b> División de la ROI para estimar el tamaño.....	64
<b>Figura 25:</b> (a) Mapa de disparidad para la detección (b) estimación del tamaño del pez (longitud obtenida 82cm; longitud real 80.9 cm).....	65
<b>Figura 26:</b> Ejecución de algoritmos en el Vídeo 1.....	67
<b>Figura 27:</b> Ejecución de algoritmos en el Vídeo 2.....	67
<b>Figura 28:</b> (a) y (c) detecciones del pez (b) y (c) se observan cómo afectan los cambios de luz y las sombras.....	68
<b>Figura 29:</b> Posición de los sensores en la técnica propuesta.....	72
<b>Figura 30:</b> Sensor láser URG-04LX-UG01.....	73
<b>Figura 31:</b> Características del sensor.....	74
<b>Figura 32:</b> Dimensiones externas del sensor.....	74
<b>Figura 33:</b> Cámara utilizada para la toma de imágenes.....	76
<b>Figura 34:</b> Imágenes durante los ensayos de medición del láser.....	76
<b>Figura 35:</b> Captura de pantalla del software <i>UrgBenri</i> .....	77
<b>Figura 36:</b> Mediciones realizadas para analizar el comportamiento del láser en agua utilizando los niveles de turbidez (a) agua limpia, (b) nivel bajo-medio, (c) nivel medio-alto y (d) nivel alto.....	78
<b>Figura 37:</b> Cálculo de distancias mediante un sensor láser.....	79
<b>Figura 38:</b> Detalle de las diferentes etapas en el proceso de visión artificial. (a) imagen generada (b) imagen después del pre-procesamiento (c) imagen al término del proceso de segmentación (d) imagen con los objetos candidatos seleccionados.....	80
<b>Figura 39:</b> Procesamiento del buffer de lecturas enviadas por el sensor cuando recibe una nueva entrada.....	81
<b>Figura 40:</b> Esquematización de la primera fase de la técnica propuesta.....	83
<b>Figura 41:</b> Esquematización del algoritmo utilizado.....	85
<b>Figura 42:</b> Proceso de seguimiento de un pez.....	86
<b>Figura 43:</b> Pantalla principal de la aplicación.....	87
<b>Figura 44:</b> Mapa de la pantalla principal.....	88
<b>Figura 45:</b> Zona de parámetros láser.....	88
<b>Figura 46:</b> Menú de la sección “Parámetros Láser”.....	90
<b>Figura 47:</b> Pantalla de modificación de parámetros del láser.....	90
<b>Figura 48:</b> Menú de parámetros del programa.....	91
<b>Figura 49:</b> Pantalla de modificación de parámetros de la identificación y seguimiento.....	92

<b>Figura 50:</b> (a) Escala de metacrilato y (b) montaje de la cámara y el láser.....	93
<b>Figura 51:</b> Resultado del proceso de visión artificial sobre la imagen capturada por la cámara.....	94
<b>Figura 52:</b> Kinect versión 1.....	98
<b>Figura 53:</b> Kinect versión 2.....	98
<b>Figura 54:</b> Dispositivo Time of Flight .....	99
<b>Figura 55:</b> Nube de puntos que genera la Kinect .....	100
<b>Figura 56:</b> Funcionamiento del buffer de frames .....	105
<b>Figura 57 :</b> Prueba con un pez que se identifica correctamente (en amarillo): (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscar máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscar mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación .....	112
<b>Figura 58:</b> Prueba con dos peces de diferente tamaño (identificados en amarillo) : (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscar máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscar mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación .....	113
<b>Figura 59:</b> Prueba con 2 peces que se solapan: (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscar máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscar mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación .....	114
<b>Figura 60:</b> Prueba con un pez de diferente forma y tamaño: (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscar máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscar mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación.....	115
<b>Figura 61:</b> Prueba con un pez de diferente forma y tamaño: (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscar máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscar mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación.....	116
<b>Figura 62:</b> Detección de regiones con movimiento (a) máscara, (b) contornos de la máscara, (c) contornos válidos .....	117
<b>Figura 63:</b> Región con movimiento (a) Identificación de la región, (b) Profundidad de la región filtrada .....	118
<b>Figura 64:</b> Detección de bordes (a) máscara de bordes, (b) puntos seleccionados para la unión de bordes.....	118

<b>Figura 65:</b> Siluetas candidatas que pueden representar uno o dos peces .....	119
<b>Figura 66:</b> Prueba de detección de siluetas con peces de 3.5 cm de longitud.....	119
<b>Figura 67:</b> Detección y extracción de siluetas (a) infrarrojo, (b) profundidad, (c) siluetas extraídas.....	120
<b>Figura 68:</b> Proceso de fusión de partes de siluetas y detección del pez ocluido .....	121
<b>Figura 69:</b> Seguimiento de dos siluetas en frames consecutivos por medio de la minimización de la distancia de sus puntos extremos (en amarillo) .....	123
<b>Figura 70:</b> Extracción de elipse (a) contorno, (b) elipse, (c) extremos de la elipse ....	123
<b>Figura 71:</b> A partir de los puntos extremos (a) se calculan las distancias euclídeas (b) .....	124
<b>Figura 72:</b> Frames utilizados en las pruebas con puntos extraídos automáticamente (marcados en amarillo) (a) frame 1, (b) frame 2, (c) frame 3, (d) frame 4.....	125
<b>Figura 73:</b> Frames utilizados en las pruebas con puntos extraídos automáticamente (a) frame 1, (b) frame 2, (c) frame 3, (d) frame 4 .....	126
<b>Figura 74:</b> Ejemplo de pantalla del software de visualización en funcionamiento. ....	128
<b>Figura 75:</b> Ejemplo de pantalla con la detección de 2 peces y la obtención de su tamaño. ....	128
<b>Figura 76:</b> Ejemplo de detección de dos peces de diferente tamaño en condiciones de baja luminosidad.....	129
<b>Figura 77 :</b> Imagen de uno de los tanques del Aquarium Finisterrae.....	130
<b>Figura 78 :</b> Imagen de uno de los tanques del Aquarium Finisterrae.....	131
<b>Figura 79 :</b> Imagen de uno de los tanques del Aquarium Finisterrae.....	131
<b>Figura 80:</b> Imagen de uno de los tanques del Aquarium Finisterrae.....	132
<b>Figura 81:</b> Ejemplo de detección en la herramienta en un tanque con peces reales ...	133
<b>Figura 82:</b> Ejemplo de detección en la herramienta en un tanque con peces reales ...	133
<b>Figura 83:</b> Ejemplo de detección en la herramienta en un tanque con peces reales ...	134
<b>Figura 84:</b> Ejemplo de detección en la herramienta en un tanque con peces reales ...	134

# 1. Resumen largo

## 1.1. Castellano

En esta tesis se presenta una metodología de detección, seguimiento y medición de peces bajo el agua utilizando técnicas de teledetección. En la mayoría de las aplicaciones en las que se precisa la detección del pez y la medida de su tamaño, es imprescindible que se encuentren en su hábitat nadando libremente y sin ningún dispositivo eléctrico o mecánico colocado en el propio pez. De esta manera no se condiciona su comportamiento ni se somete a estrés que pudiera desencadenar situaciones no deseadas. Las técnicas de teledetección basadas en luz visible e infrarroja permiten detectar el pez sin contacto físico, sin embargo, es necesario adecuarlas para su correcto funcionamiento bajo el agua. En esta tesis se presenta esta metodología y tres aplicaciones prácticas de dispositivos de detección basados en principios físicos diferentes. La primera técnica se basa en cámaras de visión sumergibles, en este caso se desarrolla un software que permite el pre-procesado de este tipo de imágenes donde el agua distorsiona la visión, hay presencia de burbujas, poca luminosidad, reflejos, etc. También es necesario el desarrollo de algoritmos que permiten la detección del pez en estas condiciones y el posterior procesado para calcular el tamaño del pez mediante técnicas de visión artificial basadas en visión estereoscópica. Todas estas técnicas existen para el tratamiento de imágenes fuera del agua, pero es preciso su adaptación al funcionamiento bajo el agua. La segunda técnica se basa en detección de objetos mediante luz infrarroja con dispositivos denominados escáner 2D. Estos dispositivos proporcionan un vector de distancias de objetos detectados a diferentes ángulos del sensor, y están pensados para su funcionamiento fuera del agua, como detección de objetos en robótica. En este caso se aplica la metodología para adaptar esta información a objetos bajo el agua, con su calibración y traslación a pseudo imágenes para la detección del paso de un pez enfrente del sensor. También se desarrollan los algoritmos de detección y medición del pez para su funcionamiento real bajo el agua. La tercera técnica se basa también en luz infrarroja, pero en este caso mediante luz estructurada bajo un patrón de puntos calibrado para la detección de las deformaciones del dicho patrón según la refracción de la luz infrarroja en los objetos cercanos del sensor. Estos dispositivos denominados escáner 3D obtienen

una matriz de distancias de los objetos cercanos en la forma de una imagen. Se aplica la metodología propuesta en esta tesis para la obtención de unos algoritmos que permiten adaptar el funcionamiento de estos dispositivos en detección de objetos bajo el agua. Se mostrarán los resultados que se obtienen en ensayos reales de detección de bancos de peces con la medición de múltiples peces de forma simultánea y las aplicaciones reales donde poder aplicar estas técnicas con estos dispositivos hardware.

Las tres aplicaciones prácticas propuestas en esta tesis se muestran siguiendo las etapas metodológicas establecidas, que servirán como base para demostrar la viabilidad de las técnicas de teledetección de los dispositivos utilizados para la detección del pez bajo el agua y la validación de la metodología propuesta en el desarrollo de algoritmos adaptados a la detección de objetos en movimiento bajo el agua. Mediante esta tesis se pretende indagar en la viabilidad de utilizar dispositivos, inicialmente diseñados para funcionar en entornos fuera del agua, que pueden ser adaptados mediante algoritmos informáticos en el seguimiento de objetos bajo el agua, y principalmente en la detección y medición de peces en su entorno acuático.

## **1.2. Galego**

Nesta tese preséntase unha metodoloxía de detección, seguimento e medición de peixes baixo a auga utilizando técnicas de teledetección. Na maioría das aplicacións nas que se precisa a detección do peixe e a medida do seu tamaño, é imprescindible que se atopen no seu hábitat nadando libremente e sen ningún dispositivo eléctrico ou mecánico colocado no propio peixe. Deste xeito non se condiciona o seu comportamento nin se somete a estrés que puidese desencadear situacións non deseadas. As técnicas de teledetección baseadas na luz visible e infravermella permiten detectar o peixe sen contacto físico, con todo, é preciso adecualas para o seu correcto funcionamento baixo a auga. Nesta tese preséntase esta metodoloxía utilizando visión artificial e tres aplicacións prácticas con dispositivos de detección baseados en principios físicos diferentes. A primeira técnica baséase en cámaras de visión somerxibles, neste caso desenvólvese un software que permite o pre-procesado deste tipo de imaxes onde a auga distorsiona a visión, hai presenza de burbullas, pouca luminosidade, reflexos, etc. Tamén é necesario

o desenvolvemento de algoritmos que permiten a detección do peixe nestas condicións e o posterior procesado para calcular o tamaño do peixe mediante técnicas de visión artificial baseadas en visión estereoscópica. Todas estas técnicas existen para o tratamento de imaxes fora da auga, pero é preciso a súa adaptación ao funcionamento baixo a auga. A segunda técnica baséase en detección de obxectos mediante luz infravermella con dispositivos denominados escáner 2D. Estes dispositivos proporcionan un vector de distancias de obxectos detectados a diferentes ángulos do sensor, e están pensados para o seu funcionamento fora da auga, como detección de obxectos en robótica. Neste caso aplícase a metodoloxía para adaptar esta información a obxectos baixo a auga, coa súa calibración e translación a pseudo imaxes para a detección do paso dun peixe enfronte do sensor. Tamén se desenvolven os algoritmos de detección e medición do peixe para o seu funcionamento real baixo a auga. A terceira técnica baséase tamén en luz infravermella, pero neste caso mediante luz estruturada baixo un patrón de puntos calibrado para a detección das deformacións do devandito patrón segundo a refracción da luz infravermella nos obxectos próximos do sensor. Estes dispositivos denominados escáner 3D obteñen unha matriz de distancias dos obxectos próximos na forma dunha imaxe. Aplícase a metodoloxía proposta nesta tese para a obtención duns algoritmos que permiten adaptar o funcionamento destes dispositivos en detección de obxectos baixo a auga. Mostraranse os resultados que se obteñen en ensaios reais de detección de bancos de peixes coa medición de múltiples peces de forma simultánea e as aplicacións reais onde poder aplicar estas técnicas con estes dispositivos hardware.

As tres aplicacións prácticas propostas nesta tese móstranse seguindo as etapas metodolóxicas establecidas, que servirán como base para demostrar a viabilidade das técnicas de teledetección dos dispositivos utilizados para a detección do peixe baixo a auga e a validación da metodoloxía proposta no desenvolvemento de algoritmos adaptados á detección de obxectos en movemento baixo a auga. Mediante esta tese preténdese indagar na viabilidade de utilizar dispositivos, inicialmente deseñados para funcionar en contornas fora da auga, que poden ser adaptados mediante algoritmos informáticos no seguimento de obxectos baixo a auga, e principalmente na detección e medición de peixes na súa contorna acuática.

Tamén se pretende indagar na viabilidade de utilizar dispositivos inicialmente deseñados para funcionar en contornas fora da auga, que poden ser adaptados mediante

algoritmos informáticos no seguimento de obxectos baixo a auga e, principalmente, na detección e medición de peixes na súa contorna acuática.

## 1.3. English

This PhD thesis presents a methodology to underwater fish detection, tracking and measuring, using remote sensing techniques. In most cases the applications where fish detection and size measurement is required, it is essential that they are in their hábitat, swimming freely and without any electrical or mechanical device attached to the fish itself. In this way, their behavior is not conditioned or they are subjected to stress that could cause unwanted situations. Remote sensing techniques based on visible light and infrared light allow the fish to be detected without physical contact however, it is necessary to adapt them for their correct operation under water. In this thesis, this methodology using computer vision and three practical applications with detection devices based on different physical principles are presented. The first technique is based on waterproof cameras and a software is developed in order to allow the pre-processing of images where water distorts the vision, there are bubbles, low light, reflections, etc. It is also necessary to develop algorithms that allow the detection of the fish in these conditions and then the processing to calculate the fish size using computer vision techniques based on stereoscopic vision. All these techniques already exist for processing images out of the water, but their adaptation to underwater operation is necessary.

The second technique is based on object detection using infrared light with 2D scanners, which provide a vector of distances of detected objects at different angles of the sensor. These 2D scanners are intended for operation out of the water, such as object detection in robotics. In this case, the methodology is applied to adapt this information to underwater objects, with its calibration and translation to pseudo images for the detection of the passage of a fish in front of the sensor. Fish detection and measurement algorithms are also developed for their real operation under water.

The third technique is also based on infrared light, but in this case by means of structured light under a pattern of points which is calibrated for the detection of

deformations of that pattern according to the refraction of infrared light in objects close to the sensor. These devices called 3D scanners obtain a matrix of distances of nearby objects as an image. The methodology proposed in this thesis is applied to obtain algorithms that allow adapting the operation of these devices in detecting underwater objects. The results obtained in real tests for the detection of schools of fish with the measurement of multiple fish simultaneously and the real applications where these techniques can be applied with these hardware devices will be shown.

The three practical applications proposed in this thesis are explained following the established methodological steps. In this way, they will serve as a basis to demonstrate the feasibility of remote sensing techniques of the devices used for the detection of fish underwater and the validation of the proposed methodology in the development of algorithms adapted to the detection of moving objects underwater. Through this thesis it is intended to investigate the feasibility of using devices, initially designed to work in outside the water environments, which can be adapted by means of computer algorithms in the tracking of underwater objects, and mainly in the detection and measurement of fish in their environment.

## 2. Introducción

La detección de objetos es un campo de aplicación ampliamente estudiado en el ámbito de la visión artificial, una buena identificación del objeto es la parte fundamental que proporciona la base para recopilar información a través del procesamiento de imágenes en un sistema automático. En esta tesis, el objeto a detectar será un pez, en su hábitat natural, sumergido en el agua donde las condiciones de luz y visibilidad pueden ser muy variables. La detección de peces en su entorno natural mediante visión artificial permite evitar técnicas potencialmente invasivas para los peces, como el uso de sensores [1], someter los peces a estrés u otras técnicas tradicionales como la visión directa que implica la supervisión continua por parte de un operador. En esta tesis se hace una revisión de las técnicas que pueden aplicarse para detectar el pez sin contacto directo y de la manera que menos se invada el hábitat del pez. El poder identificar un pez bajo el agua permite realizar varias aplicaciones muy interesantes, que van desde el seguimiento del crecimiento de los peces en la acuicultura, con el fin de adaptar su alimentación, a la estimación de las diferencias en el tamaño del cuerpo del pez, siendo posible estimar el crecimiento de estos a lo largo del tiempo. Otra aplicación posible es el control del paso de los peces por un río, donde se puede estimar el tamaño del pez y el número de peces que han pasado en un período de tiempo.

El uso de cámaras de vídeo es uno de los métodos más potentes para detectar objetos, ya que proporciona una gran cantidad de información y también es uno de los métodos más económicos. Para detectar objetos bajo el agua hay diferentes técnicas, una ampliamente utilizada es la tecnología acústica [2, 3]. Una de las grandes ventajas de la tecnología acústica es la posibilidad de detectar objetos a muchos rangos de distancia, pero la mayor limitación es obtener o detectar el tamaño real del objeto. Otra limitación es la velocidad de adquisición, ya que los dispositivos requieren un tiempo de transmisión y procesamiento de señal que limita su aplicación a objetos que no se mueven rápidamente. La tecnología acústica se utiliza para detectar bancos de peces, peces grandes o peces de movimiento lento [4]. Sin embargo, si el objetivo es detectar un pez individual, de tamaño pequeño o mediano, y también es necesario conocer su tamaño, la tecnología acústica no es la más adecuada. Usando cámaras de video, es posible obtener mucha información y a alta velocidad. El principal problema es que bajo el agua la imagen

se deteriora bastante, debido a numerosos factores, como la turbidez, las partículas flotantes, las burbujas y la atenuación de la luz a diferentes profundidades. Ha habido investigaciones que tratan de restaurar o mejorar la calidad de las imágenes obtenidas bajo el agua utilizando diferentes técnicas [5-7]. Debido a la poca visibilidad y a los efectos no deseados que se obtienen en las imágenes, la detección de objetos mediante cámaras en entornos acuáticos se considera una tarea difícil. Las técnicas de visión artificial para el procesamiento de imágenes submarinas están en continuo desarrollo, principalmente en robótica, para el desarrollo de la navegación autónoma de robots submarinos [8]. El objetivo es detectar peces que suelen estar en movimiento, incluso a gran velocidad, y una vez detectados, estimar su tamaño. Para ello, existen técnicas como la utilización de una cámara estereoscópica bajo el agua, de la que se obtienen dos vídeos a velocidades de 25-30 fotogramas por segundo, lo que permite capturar los peces incluso en movimiento. Sin embargo, también es posible utilizar otros rangos del espectro que se encuentren fuera de la zona visible, como es el caso del infrarrojo. Existen dispositivos que permiten detectar y medir distancias utilizando esta tecnología. Aunque su principal campo de aplicación es fuera del agua. Existen escáneres de objetos tanto 2D como 3D, utilizados principalmente en el ámbito de la robótica, los videojuegos y cada vez más con aplicaciones en el campo de la industria, topología, etc. En esta tesis se analizará su posible utilización en ambientes acuáticos, tanto los escáneres 2D como los escáneres de luz estructurada 3D, analizando las limitaciones que en este campo tiene y sus posibilidades de aplicación.

En esta tesis se realizarán 3 aplicaciones prácticas de detección de peces para demostrar las posibilidades de aplicación que ofrecen las tecnologías de infrarrojos y las cámaras de visión clásicas combinadas con diferentes técnicas de Inteligencia Artificial, como son la Visión Artificial y las Redes de Neuronas Artificiales. Cada técnica va acompañada de unos ensayos realizados para testar la técnica y en donde se consiguen porcentajes de detección superiores al 70%, llegando en una de ellas al 80%. Parte de estos ensayos se han realizado con peces reales en tanques cerrados. En el trabajo futuro, estos ensayos deberían tratar de realizarse en un entorno más real y menos predecible para evaluar más en profundidad estas técnicas.

## 2.1. Estructura del documento

Esta tesis está organizada en capítulos de la siguiente manera:

*Capítulo 3:* revisión al campo de la visión artificial aplicada a la detección de peces, con la evolución de la tecnología de adquisición de imágenes desde los sistemas 2D hasta los 3D y la utilización de software que permite el uso de técnicas de visión artificial, de inteligencia artificial o de *Deep learning*. Se revisan también otros sistemas como los contadores de peces, sobre los que se pueden aplicar las investigaciones de esta tesis.

*Capítulo 4:* se define la hipótesis de partida y se establece la metodología que se desarrollará en cada caso práctico.

*Capítulo 5:* desarrollo del primer caso práctico consistente en una aplicación de detección de peces basada en la búsqueda del ojo del pez con Visión Artificial y Redes de Neuronas Artificiales, en el que se va desarrollando la metodología comentada, los ensayos llevados a cabo y se termina con unas conclusiones.

*Capítulo 6:* desarrollo del segundo caso práctico consistente en la detección de peces en escala de hendidura vertical combinando técnicas de visión artificial con la tecnología láser. Al igual que en el caso anterior se va explicando la metodología llevada a cabo, posteriormente los resultados de los ensayos experimentales y por último se ofrecen unas conclusiones.

*Capítulo 7:* Se explica la metodología llevada a cabo para este último caso práctico en el que se desarrolla un sistema para la detección de peces mediante la luz estructurada. Se presentan también las conclusiones extraídas tras ver los resultados obtenidos en los ensayos realizados.

*Capítulo 8:* Conclusiones finales y trabajo futuro a desarrollar como continuación a esta tesis.

*Capítulo 9:* Publicaciones realizadas durante el desarrollo de esta tesis.

### 3. Estado del arte

La importancia de la detección y medición de los peces se ha estudiado para diversos escenarios como por ejemplo el comportamiento de peces en los ríos o el estudio de los especímenes en las piscifactorías. En el primer caso, en los ríos existen unas estructuras hidráulicas, como las escalas de hendidura vertical (Figura 1), que permiten el movimiento de los peces para sortear obstáculos (presas, centrales hidroeléctricas, etc.). Se sabe que el diseño de una escala de peces de hendidura vertical depende de la interacción de parámetros hidráulicos y biológicos, ya que las variables hidrodinámicas de la piscina (como la velocidad del agua) deben ser adecuadas a los requisitos de los peces (determinados por ejemplo por su resistencia o velocidad de natación). Se ha estudiado el flujo en las escalas de hendidura vertical [9-11], la natación de los peces [12, 13] y el comportamiento de los peces dentro de la escala [14]. En cuanto a las piscifactorías, la importancia de conocer la evolución de los tamaños de cada pez para detectar problemas o saber el momento idóneo de la extracción, repercuten directamente en el negocio.



**Figura 1:** Estructura de una escala de hendidura vertical, ubicada en el Centro de Estudios y Experimentación de Obras Públicas (CEDEX) del Centro de Estudios Hidrográficos (CEH)

La detección de objetos, en general, ha sido un tema muy discutido en los estudios de visión artificial, ya que una buena identificación del objeto requerido proporciona la

base para recopilar información mediante el procesamiento de imágenes en un sistema de este tipo.

Existen multitud de sensores y dispositivos que permiten recopilar la información necesaria para, posteriormente, aplicar las operaciones de visión artificial que permitan detectar los objetos.

Uno de los medios de adquisición más utilizados y que más han evolucionado son las cámaras. La capacidad de realizar cada vez sensores más pequeños y a la vez con mejor resolución ha supuesto un gran avance.

La información de las cámaras se puede combinar con información obtenida por algún otro medio de adquisición de datos para conseguir sistemas más precisos o incluso utilizar estos medios de forma aislada. Estos dispositivos pueden recoger información en 2 dimensiones o 3 dimensiones.

Entre los medios de 2 dimensiones se encuentran los sensores láser, el LIDAR, los infrarrojos, acústicos, térmicos, etc. De hecho, existen combinaciones comerciales que integran tanto el sensor de cámara como un segundo sensor, dotando al sistema de mayor resolución y mejor definición que si se usan de forma separada. Entre ellas se pueden encontrar, cámaras de alta velocidad, cámaras de infrarrojos, cámaras multiespectrales, cámaras submarinas, cámaras inteligentes, cámara con láser 3D.

La visión 3D ha dado un impulso importante a la tecnología de reconocimiento de objetos. Gracias a su uso mediante la tecnología láser se pueden estudiar los objetos en el espacio de forma individual ya que facilita su segmentación. Además permite mediciones de volúmenes y posicionar exactamente cada objeto de la escena.

Los sensores 3D [15] o escáneres 3D son dispositivos que mapean la escena generando una nube de puntos que se corresponde con la superficie de los objetos que se encuentran en ella, existen diferentes clases de sensores 3D que utilizan diferentes métodos. En este trabajo se van a utilizar los que son sin contacto (utilizan el comportamiento de radiaciones en la escena) y dentro de ellos los activos. Los sensores activos son los sensores 3D que emiten algún tipo de radiación y generan la nube de

puntos a partir del reflejo de la radiación emitida en la escena, su velocidad y precisión hacen que sean los más utilizados actualmente, el principal problema es que necesitan hardware especializado y en algunos casos el coste de estos dispositivos es muy elevado. Entre este tipo de sensores se pueden nombrar los *Time of Flight* [16], los de luz estructurada [17] y los de triangulación.

A pesar de que el impulso de esta tecnología es relativamente reciente ya existen multitud de campos en los que se ha utilizado, por ejemplo en la arquitectura (en donde esta tecnología permite almacenar los esquemas de la estructura real, y que permitiría por ejemplo su reconstrucción en caso de que fuera necesario o la restauración de obras [18], en la construcción para modelos geológicos, ingeniería civil [19], entre otros campos de la construcción [20], en la industria de la automoción [21], para determinar posiciones y formas de cuerpo humano [22] aplicaciones en la naturaleza para los bosques como en [23] o para el seguimiento de múltiples peces como en [24]. También existen estudios con el sensor concreto que se va a utilizar para la detección de objetos [25] o de personas [26], aunque su aplicación al medio submarino es menos habitual, en donde las situaciones muy cambiantes tanto ambientales como de luminosidad complican el escenario de manera considerable.

Las técnicas de visión artificial para el procesamiento de imágenes, muy utilizadas por ejemplo en robótica para el desarrollo de la navegación autónoma de robots submarinos [8], pueden aplicarse al campo de la detección de peces en su medio natural. En [27] se detallan muchos de los principales modelos utilizados para la detección de peces en la acuicultura. Para aplicar las técnicas de visión artificial se necesita hacer uso de grabaciones de imágenes que se pueden recabar utilizando diferentes tecnologías. Esta aplicación de la visión artificial permite dejar de lado técnicas potencialmente invasivas para los peces, como el uso de sensores [1], que someten a los ejemplares a estrés, u otras técnicas tradicionales como la visión directa que implica la supervisión continua por parte de un operador.

El uso de cámaras de vídeo es uno de los métodos más poderosos para detectar objetos, ya que proporciona una gran cantidad de información a alta velocidad y también es uno de los métodos más baratos. En [28] se utilizan vídeos sincronizados (28 cámaras en total, repartidas en bloques de 4) para, a través de algoritmos y técnicas de visión

artificial, desarrollar una técnica que automáticamente detecta, realiza el seguimiento y analiza el comportamiento y la trayectoria que siguen los peces dentro de las escalas de los ríos, obteniendo un 95% de precisión. Estos resultados permiten establecer guías futuras para el diseño de las escalas de hendidura vertical.

El principal problema es que bajo el agua la imagen se deteriora mucho, debido a numerosos factores, como la turbidez, las partículas flotantes, las burbujas y la atenuación de la luz a diferentes profundidades. Se han realizado investigaciones que intentan restaurar o mejorar la calidad de las imágenes obtenidas bajo el agua utilizando diferentes técnicas [5-7]. En [29] los autores usan visión artificial para detectar peces mediante vídeos submarinos de baja visibilidad, utilizando métodos de sustracción de fondo, filtrando las imágenes y una técnica de post-procesamiento, obteniendo una precisión del 60%.

Debido a la escasa visibilidad y los efectos no deseados, la detección de objetos mediante cámaras en entornos submarinos se considera una tarea desafiante. Además, la mayoría de las técnicas utilizadas en visión artificial requieren un conocimiento previo de las especies de peces a detectar [30, 31] y del fondo donde se capturará la imagen, lo que restringe las posibilidades de detección en un escenario más abierto y real. Por lo tanto la detección se puede realizar utilizando redes neuronales [14], *snakes* [32], ajustes de nivel [31], etc. En otros casos se opta por aplicar las técnicas de visión artificial en imágenes con los peces fuera del agua, por ejemplo en [33] se hace para medir los peces.

Además del uso de cámaras, dos de las tecnologías más utilizadas para la detección de peces es la tecnología acústica y el láser.

Dentro de la tecnología acústica, el sónar es un aparato que detecta la presencia y situación de objetos sumergidos mediante ondas acústicas. Desde los años sesenta esta técnica se ha usado en aplicaciones como un contador de peces [34] o para obtener características de los peces [35]. Más recientemente apareció el *Dual-frequency identification sonar* (DIDSON), sónar de identificación de doble frecuencia que reduce las limitaciones de los sistemas acústicos anteriores y consigue imágenes de más calidad, y con ella nuevos estudios para detección y conteo de peces aprovechando la tecnología acústica [36-39].

Una de las grandes ventajas de la tecnología acústica es la posibilidad de detectar objetos a muchos rangos de distancia, pero la mayor limitación es obtener o detectar el tamaño real del objeto. Otra limitación es la velocidad de adquisición, ya que los dispositivos requieren un tiempo de transmisión y procesamiento de señal que limita su aplicación a objetos que no se muevan rápidamente. La tecnología acústica se utiliza para detectar bancos de peces, peces grandes o peces de movimiento lento [4]. Sin embargo, si el objetivo es detectar un pez individual, de tamaño pequeño o mediano, y además es necesario conocer su tamaño, la tecnología acústica no es la más adecuada.

La otra tecnología comentada es la del láser de infrarrojos. El láser de infrarrojos proyecta un haz de luz infrarroja que permite la detección de objetos cuando este haz es reflejado al ser interrumpido por el objeto y permite conocer a la distancia que se encuentra el objeto. Este tipo de sensores en el medio acuático permiten realizar detección de peces [40] en condiciones lumínicas no adecuadas para el ojo humano o para los sensores de las cámaras, ya que indicarán la detección del objeto y, correctamente calibrado, la distancia a la que se encuentra.

Una de las aplicaciones comerciales de esta técnica (que para mayor utilidad se complementa con otras herramientas, como cámaras de vídeo), y sobre la que se trabaja en esta tesis es la del contador de peces. Uno de los contadores de peces más conocidos es el *Riverwatcher Fish Counter* [41] (Figura 2) de la empresa islandesa *Vaki* [42], que utilizando un escáner de infrarrojos y combinándolo con cámaras de alta resolución y un túnel con leds permite la toma de fotos y videos para obtener así información del pez detectado (Figura 3). La unidad de escáner consiste en dos placas de escáner distanciadas entre ellas entre 10 cm y 45 cm, en donde los diodos de luz envían rayos de luz infrarroja a los receptores del otro lado. En el momento en el que un pez nada a través de la red de rayos de luz, la imagen de la silueta resultante se usa para contar y estimar el tamaño de cada pez. Además esa imagen se almacena para poder ser luego verificada. La unidad de escáner puede unirse con un sistema que mediante una cámara permite grabar vídeo o capturar el momento del paso del pez a través del escáner. Los datos aportados por la empresa señalan una precisión en el conteo de peces del 98% y una precisión en la medición de cada pez mayor del 95%.



**Figura 2 :** Estructura del dispositivo Riverwatcher Vaki [42]

Actualmente la empresa Vaki ha sacado al mercado un nuevo modelo que consta de una cámara de vídeo IP estéreo con visión de luz infrarroja, luces LED blancas e infrarrojas, un túnel de acero, para controlar que la calidad de la imagen y el vídeo sea óptimo y las condiciones de luz sean estables, y un ordenador de alto rendimiento. Además el sistema permite la visualización en tiempo real en *streaming*, pudiéndose ver desde cualquier dispositivo conectado a internet.



**Figura 3:** Cámara con leds utilizado por Vaki en el Riverwatcher [42]

Además del *Riverwatcher Fish Counter* existen algunas otras soluciones comerciales utilizadas para el conteo de peces como *AquaScan Fishcounters* (Figura 4)

de la empresa noruega *AquaScan* [43], que realiza el conteo de peces transportados a través de tuberías o canales planos, haciendo uso de leds infrarrojos.



**Figura 4 :** Uno de los contadores de AquaScan, en este caso el CSW27600 [43]

También utiliza leds infrarrojos para este conteo de los peces transportados por tuberías o canales la empresa *Calitri Technology* [44] en sus diferentes contadores (Figura 5). Misma tecnología que utiliza también la empresa *Faivre* [45] (Figura 6).



**Figura 5:** Contador de peces de *Calitri Technology* y un posible montaje [44]



**Figura 6:** Dos modelos de contadores de la empresa Faivre [45]

Otras empresas con solución comercial para el conteo de peces aunque diferente tecnología a la de infrarrojos son *Impex Agency ApS* [46] con su contador TPS (Figura 7) (en este caso para peces pequeños) o *Smith-Root* [47] con su contador (Figura 8) basado en la conductividad eléctrica según la cual el paso de peces por el túnel, causará cambios en la conductividad del agua.



**Figura 7:** Contador de peces de Impex Agency ApS [46]



**Figura 8:** Unidad del contador de peces SR-1601 de Smith-Root [47]

Todas estas aplicaciones comerciales requieren de una gran inversión y de unas estructuras fijas de grandes dimensiones y pesos, ya sea con su sección fija necesaria, como en el Vaki, o con la infraestructura que acompaña al detector, en el caso de los que necesitan tuberías, túneles o canales, así como de una zona controlada para su ubicación.

Para trabajar con las técnicas de visión artificial, han surgido herramientas software (librerías) que permiten mayor comodidad y eficiencia. Sherlock [48] es una librería comercial que basa su funcionamiento en una interfaz gráfica para desarrollar las técnicas. Otra librería con licencia comercial es Halcon [49], pensada especialmente para entornos industriales y que cuenta con su propio lenguaje de programación. Dispone también de una interfaz gráfica para un desarrollo más visual. A pesar de contar con su propio lenguaje, permite programar en otros como C, C++, C#, entre otros. Es una librería muy versátil ya que permite desarrollar aplicaciones de visión artificial tanto a un usuario con pocos conocimientos como a un experto.

Entre las librerías open-source, destacan Point Cloud Library [50], más orientada al mundo 3D y que lanzó su primera versión en 2011, y sobre todo OpenCV [51] que es una de las librerías más utilizadas en el mundo de la visión artificial. Está desarrollada en C y C++ y es multiplataforma. Está publicada bajo licencia BSD. Originalmente fue desarrollada por INTEL en 1999 y actualmente tiene una gran cantidad de algoritmos, que incluye algoritmos de machine learning y de visión artificial. Estos algoritmos permiten identificar caras, objetos, extraer modelos 3D, eliminar ojos rojos, reconocer escenarios, etc. Entre sus ventajas destacan la continua actualización e incorporación de

nuevos algoritmos, la gran comunidad que tiene a su alrededor y su buena documentación. En su contra podría señalarse que su utilización es compleja para aquellos que no tengan unos buenos conocimientos de visión artificial y de programación.

La estructura de la librería es modular, cuyos módulos principales son:

- Core: Este es el módulo básico de OpenCV. Incluye las estructuras de datos básicas y las funciones básicas de procesamiento de imágenes. Este módulo también es usado por otros módulos como highgui.
- Highgui: Este módulo provee interfaz de usuario, códecs de imagen y vídeo y capacidad para capturar imágenes y vídeo, además de otras capacidades como la de capturar eventos del ratón...etc.
- Imgproc: Este módulo incluye algoritmos básicos de procesado de imágenes, incluyendo filtrado de imágenes, transformado de imágenes, operaciones morfológicas, transformaciones geométricas, etc.
- Video: Contiene la estimación de movimiento y, funciones y clases para el seguimiento de objetos y la extracción de fondos.
- Features2d: Es un módulo con el que se consigue la detección de puntos y descriptores y además posee un framework para la unión de esos puntos.
- Calib3d: Módulo que se encarga de la calibración de la cámara, se encarga de la estimación de la geometría para vistas estereoscópicas.
- ML: Contiene los componentes de la librería que están relacionados con el aprendizaje máquina, como por ejemplo algoritmos relacionados con los Support Vector Machines (SVM), Redes de Neuronas Artificiales (RNA), algoritmos de búsqueda como el KNN (K Nearest Neighbour).

Esta librería dispone de librerías envoltorio (wrapper) para determinados lenguajes. Por ejemplo, EmguCV es su versión para .NET. De esta manera se permite llamar a las funciones de OpenCV desde lenguajes compatibles con .NET (C#, Visual Basic,...).

.NET Framework [52] es un entorno de ejecución administrado que proporciona diversos servicios a las aplicaciones en ejecución. Consta de dos componentes principales: Common Language Runtime (CLR), que es el motor de ejecución que controla las aplicaciones en ejecución, y la biblioteca de clases de .NET Framework, que proporciona una biblioteca de código probado y reutilizable al que pueden llamar los desarrolladores desde sus propias aplicaciones.

EmguCV [53] tiene 2 capas, la capa básica contiene las funciones, estructura y mapas de enumeración que reflejan directamente lo que hay en OpenCV. La segunda capa contiene clases que combinan las ventajas del entorno .NET

Además de las técnicas de visión artificial comentadas anteriormente, en los últimos años, los avances que se han ido sucediendo en el campo de la inteligencia artificial y el aumento de las posibilidades de acceso a altas capacidades de cómputo han propiciado una mayor aplicación de técnicas de inteligencia artificial, como el uso de las redes de neuronas profundas o *Deep Learning*, que se empezaron a utilizar también en resolución de tareas relativas a los peces [54] y que hoy centra gran parte de los trabajos en esta línea de investigación, especialmente en la detección e identificación de especies y no tanto en el conteo o el análisis del comportamiento del pez y sus trayectorias.

Las redes convolucionales (CNN) son de las redes de *Deep Learning* más utilizadas para el tratamiento de imágenes, debido a que están diseñadas para operar en datos que con algún tipo de secuencialidad (como con los píxeles de las imágenes). El uso de estas redes hace también mejorar las técnicas de detección de objetos en tiempo real, surgiendo nuevos algoritmos como YOLO (You Only Look Once) [55], que básicamente utiliza la idea de dividir la imagen de entrada en varios grids, llevando a cabo las detecciones y clasificaciones dentro de cada grid con redes convolucionales.

En 2017, en [56] se proponía el uso de las CNN para la detección de peces consiguiendo una alta precisión. Posteriormente se han ido desarrollando multitud de trabajos en este campo, a continuación se muestran algunos de ellos. En [57] se usa una red neuronal artificial de convolución para detectar peces en agua limpia y con buena luminosidad. En [58] se utiliza la red de convolución entrenada con imágenes de peces fuera del agua para conseguir mediciones de tamaños de peces. También en [59] se hace uso del *Deep Learning*, combinando el uso de la visión estéreo con un par de redes neuronales, una red neuronal de convolución para detectar el pez y una segunda red neuronal para detectar puntos clave específicos para estimar la postura y longitud del pez, consiguiendo una tasa de error de medición de la longitud del pez cercana al 6%. En [60] también se combina la visión estéreo con un tipo de red de convolución para obtener tamaños de peces con un 88% de precisión. En [61] se muestra otra combinación de la adquisición de imágenes mediante visión estereoscópica y las redes de neuronas para estimar el tamaño del pez. Otra combinación de técnicas diferente, es la de las redes convolucionales con otras técnicas de inteligencia artificial, como el *machine learning*, como se hace en [62] y que consigue su mejor precisión con 87,44% para uno de sus conjuntos de datos. En [63] se combina el uso del *Deep Learning*, mediante el algoritmo YOLO, con información temporal para mejorar la detección y clasificación de los peces.

El *Deep Learning* también aparece combinado con otras técnicas comentadas anteriormente, como los sensores acústicos, para conseguir mejores resultados, como en [64]. Por último, una metodología más compleja se presenta en [65], en donde se usa el *Deep Learning* con la información adquirida en la combinación de sensores ópticos y sensores acústicos.

Esta evolución de la tecnología y el uso creciente de la inteligencia artificial conlleva una tendencia hacia la sofisticación de las piscifactorías inteligentes, en donde aún existen grandes oportunidades y retos a conseguir [66].

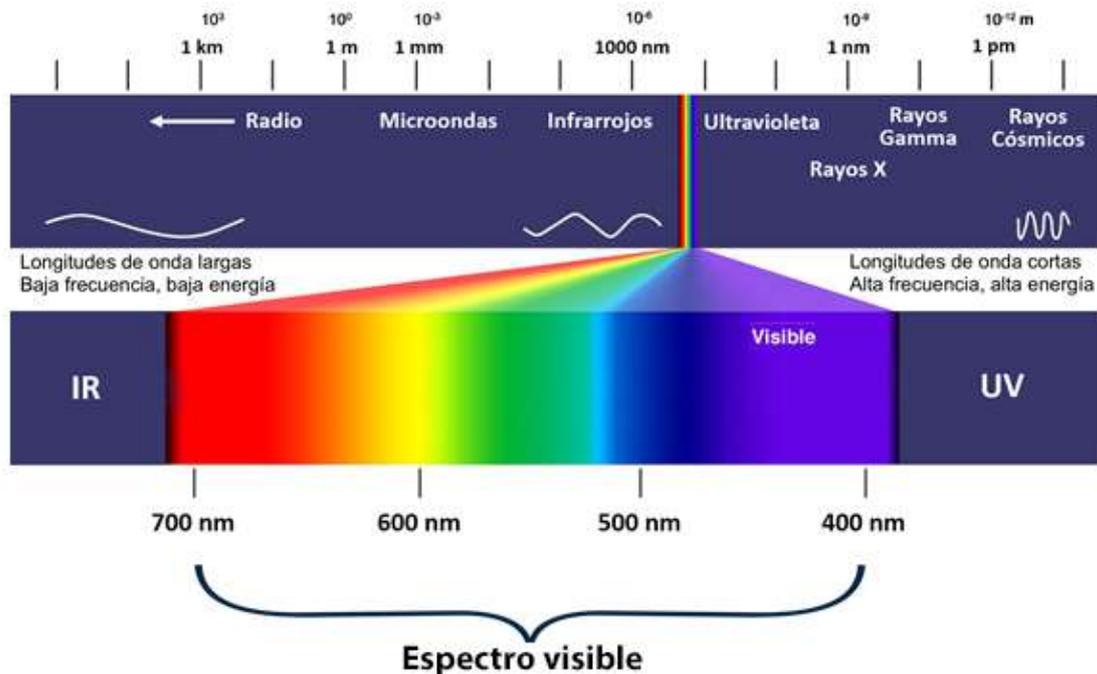
## 4. Metodología

En la presente tesis se expone una metodología de diseño de sistemas automáticos de detección de objetos en movimiento bajo el agua, especialmente enfocada a la detección de peces en su hábitat natural. Para ello se propone el uso de diferentes tecnologías en función de su aplicación práctica final.

Como **hipótesis de partida** se establece que la detección del objeto (pez) puede ser realizada con la menor intrusión posible, y siempre **sin contacto físico con el objeto**. De esta manera, para la detección de un objeto sin su contacto directo es necesario recurrir a la teledetección, esta es una técnica que permite medir variables físicas de un objeto sin el contacto físico con dicho objeto. Para ello es necesaria una interacción con el objeto a través de alguna radiación electromagnética (Figura 9). En teledetección existen principalmente tres tipos de interacción entre el sensor y el objeto:

1. Radiación solar reflejada por el objeto (puede ser luz visible, infrarrojo, ultravioleta, ondas de radio, etc.)
2. Radiación emitida por el objeto (como por ejemplo el infrarrojo para medir su temperatura)
3. Radiación emitida por el sensor y reflejada por el objeto (como por ejemplo el radar)

El uso de una cámara convencional de video se basa en el primer tipo, la luz reflejada por un objeto es capturada por el sensor de la cámara que la transforma en impulsos eléctricos. Para ello se utiliza el espectro de la luz visible y para que se refleje dicha luz debe existir una fuente de la misma. Esta fuente de luz puede proceder del sol o de fuentes artificiales como bombillas, fluorescentes, leds, etc. En todos estos casos se utiliza el espectro electromagnético que corresponde a la luz visible. Existen videocámaras que pueden utilizar parte del espectro electromagnético que está fuera de este rango, usualmente se utiliza el infrarrojo cercano.



**Figura 9:** Muestra los tipos de radiación relacionados con sus longitudes de onda y el espectro visible

Los dispositivos de medición de distancias suelen utilizar el espectro del infrarrojo para la detección del tiempo que tarda desde que se emite el impulso hasta que se recibe. Por ello, utiliza el tercer tipo de interacción en la teledetección. Dispone de una fuente de luz infrarroja que emite y luego recibe. Midiendo el tiempo que tarda puede trasladarlo a distancia. Este principio físico es el que utilizan los escáneres de objetos en 2D.

Existen dispositivos que disponen de fuente de luz infrarroja y que emiten sobre una superficie utilizando un patrón de líneas o puntos que poseen prefijados internamente. Al observar las diferencias de esos patrones, al volver la luz infrarroja reflejada, permiten extrapolar las distancias a las que se encuentran los objetos según las variaciones de ese patrón reflejado. Este funcionamiento corresponde a los escáneres 3D denominados de luz estructurada.

Estas tres tecnologías parecen las más adecuadas para la detección de objetos en movimiento, y serán las candidatas para desarrollar sistemas (hardware y software) que permitan detectar los movimientos de los peces bajo el agua. En todas ellas, será necesaria la utilización de técnicas de procesamiento en un computador, siendo preciso, en algunos

casos, aplicar técnicas de pre-procesado con filtrado de imagen, técnicas de detección en imagen y técnicas de post-procesado para el seguimiento de los peces.

Cabe resaltar que en las tres técnicas, se tendrán de una u otra manera imágenes donde se deben aplicar técnicas de visión artificial para la detección final del pez y la estimación de su tamaño. Por ello, es factible una metodología común a ellas para el desarrollo de técnicas de seguimiento de peces bajo el agua.

En estas técnicas de teledetección los pasos a seguir serían los siguientes, que pueden trasladar a etapas de la metodología propuesta en esta tesis:

1. Dispositivo hardware de captura de radiación reflejada
2. Realizar un procesado de calibración de la imagen para corregir distorsiones producidas por efectos ópticos y el efecto de la densidad del agua (diferencia en la densidad con el aire que produce menor tiempo de propagación de la radiación utilizada)
3. Transformar la información capturada a formato matriz 2D similar a una imagen
4. Realizar un pre-procesado software de esta imagen, para eliminar ruido, resaltar bordes, fusionar o eliminar rangos de datos (transformación RGB), etc.
5. Recibir en tiempo real el flujo de información como secuencia de imágenes a lo largo del tiempo, similar a un video
6. Realizar un procesado software de detección de objetos, basado en patrón o mediante aprendizaje por casos.
7. Realizar un procesado software de seguimiento de objeto en secuencia de imágenes a lo largo del tiempo
8. Realizar un procesado software para obtener mediciones del objeto, posición, tamaño, etc.
9. Producir una visualización en tiempo real en el computador y un almacenamiento de los datos procesados

Estas 9 etapas serían aplicables a las diferentes tecnologías de detección de objetos que pueden utilizarse para la detección, seguimiento y medición de un pez bajo el agua.

Según la tecnología, se hará más énfasis en algunas etapas donde es preciso adaptar la información producida por el sensor al procesamiento de información que permita la correcta detección del pez bajo el agua. Como ejemplo, en la utilización de dispositivos basados en infrarrojos para medición de distancias, donde el resultado se produce como un vector de distancias, se realizará un procesamiento software para la traslación de ese vector en el tiempo en una matriz bidimensional que se tratará como una imagen del paso de un pez por la zona de escaneo. Por ello, en todo caso, será posible la obtención de una imagen sobre la que se aplicará la metodología propuesta en esta tesis.

Los siguientes capítulos se estructuran en relación a 3 casos prácticos de aplicación de esta metodología a 3 tecnologías hardware completamente diferentes. Actualmente se pueden obtener cámaras de video con multitud de características, resolución, velocidad de captura, sensibilidad de luminosidad y que puedan ser sumergibles. Se propondrán las características que debería tener este tipo de dispositivos para poder detectar el pez bajo el agua de mejor manera posible. En relación a dispositivos escáner de objetos, existen diferentes tipos que se encuentran disponibles en el mercado. Sin embargo, para su utilización bajo el agua, no existen una gran variedad, principalmente porque la utilización de luz infrarroja tiene una baja transmisibilidad en el agua. Se propondrá la utilización de varios dispositivos con diferentes técnicas, para la detección de objetos bajo el agua. Uno de ellos se basa en medición de distancias por tiempo de tránsito, lo que producirá un vector de distancias de objetos a diferentes ángulos del sensor. Otro de ellos, se basa en la medición de la refracción de un patrón basado en puntos de los objetos cercanos del sensor. En ambos casos será necesaria una corrección de distancias debido al agua, puesto que estos dispositivos están pensados para trabajar midiendo objetos a través del aire. En esta tesis se mostrará la viabilidad de estas técnicas para la detección, seguimiento y medición de peces en diferentes casos prácticos.

# **5. Caso práctico 1: Una aplicación de detección de peces basada en la búsqueda del ojo del pez con Visión Artificial y Redes de Neuronas Artificiales**

## **5.1. Introducción**

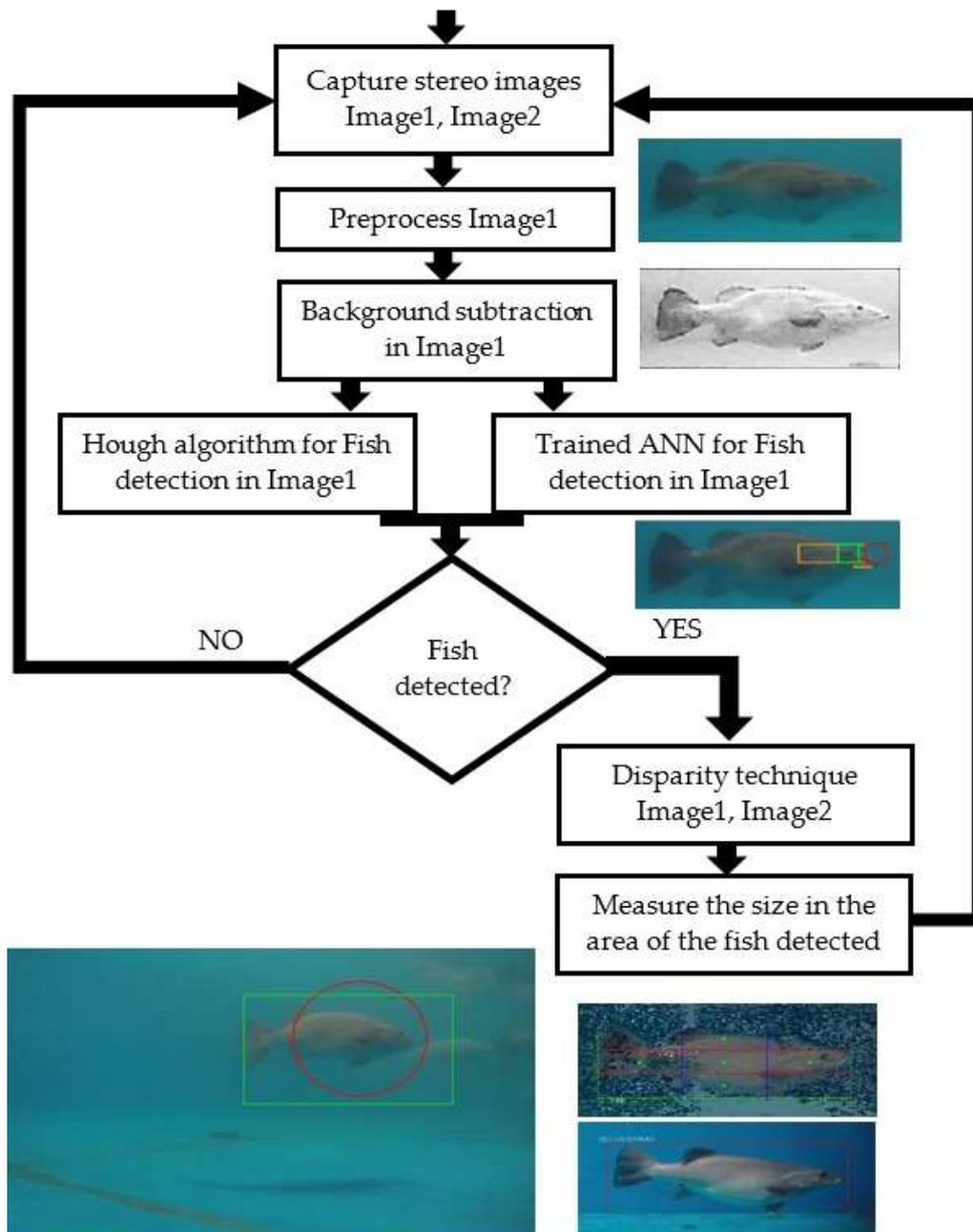
En este estudio se presenta una aplicación para la detección de peces mediante el uso de cámaras submarinas. La base de esta técnica es la búsqueda del ojo de pez en la imagen y, a partir de esta detección, poder aplicar posteriormente otras técnicas, encaminadas a conocer la forma del pez y realizar otros análisis.

La técnica propuesta se evaluó primero en un ambiente controlado, y posteriormente se realizaron experimentos en un escenario real con peces de diferente tamaño y bajo condiciones cambiantes de luz y turbidez del agua. Para la adquisición del vídeo se utilizaron dos cámaras sumergibles sincronizadas, que permiten la captura estereoscópica y así posteriormente se puede realizar una medición de los peces que hayan sido detectados.

Para poder llegar a medir la longitud o el peso de los peces, el primer paso es su detección a través de las imágenes de entrada. Para ello, se aplican diversas técnicas de filtrado de imágenes, así como modelos de sustracción de fondo y detección de contornos. Tras esos análisis y la sustracción del fondo se espera tener un resultado, con una aproximación de donde se sitúa el candidato a pez en la imagen, en donde poder aplicar una técnica de validación para asegurar que los objetos descubiertos sean realmente peces.

La primera técnica de detección de peces que se analiza se basa en la detección del ojo del pez utilizando la transformada de Hough. Paralelamente a esta técnica, se obtiene una red de neuronas artificiales (RNA) entrenada para detectar si hay un ojo de pez en la imagen y también una forma de pez. De esta forma, combinando ambas técnicas

se mejora la capacidad de detección. En la Figura 10 se muestra como sería el algoritmo llevado a cabo.



**Figura 10:** Pasos del algoritmo propuesto y ejemplos de cada paso

La idea principal de la técnica que se propone es combinar las red de neuronas con la visión artificial para detectar los peces y aumentar la precisión del proceso de detección.

## 5.2. Metodología

### 1. Dispositivo hardware de captura

Para este caso práctico se va a utilizar la cámara *GoPro Hero3+ Black* (Figura 11) con las siguientes características:

- Dimensiones cámara:
  - 5.8 x 3.9 x 2 cm
- Peso:
  - Cámara: 28 gr
  - Batería: 46 gr
  - Peso completo con carcasa: 136 gr
- Óptica
  - Tamaño sensor de imagen: 1/2.3 pulgadas de 4000 x 3000 pixeles
  - Lente 6-elemento de lente esférica de cristal ultra-fuerte
  - Apertura de f/2.8 fijo
- Grabación
  - Sistema de vídeo
    - NTSC
    - PAL
- Soporte de grabación
  - MicroSD x 1 (Clase 10 tarjeta de memoria requerido)
- Formato de vídeo
  - Códec: H.264
  - Formato: MP4

- Resoluciones de vídeo soportadas
  - 4K (16:9): 3840 x 2160 a 15 fps
  - 4K (17:9): 4096 x 2160 a 12 fps
  - 2.7K (16:9): 2704 x 1524 a 30 fps
  - 2.7K (17:9): 2704 x 1440 a 24 fps
  - Full HD (4:3): 1920 x 1440p a 48 fps
  - Full HD (4:3): 1920 x 1440p a 30 fps
  - Full HD (4:3): 1920 x 1440p a 24 fps
  - Full HD (16:9): 1920 x 1080p a 60 fps
  - Full HD (16:9): 1920 x 1080p a 48 fps
  - Full HD (16:9): 1920 x 1080p a 30 fps
  - Full HD (16:9): 1920 x 1080p a 24 fps
  - HD (4:3): 1280 x 960p a 100 fps
  - HD (4:3): 1280 x 960p a 60 fps
  - HD (4:3): 1280 x 960p a 48 fps
  - HD (16:9): 1280 x 720p a 120 fps
  - HD (16:9): 1280 x 720p a 100 fps
  - HD (16:9): 1280 x 720p a 60 fps
  - HD(16:9): 1280 x 720p a 48 fps
  - WVGA (16:9): 848 x 480p a 240 fps
  
- Conectividad
  - Conectividad inalámbrica Wi-Fi
  
- Batería
  - 1180 mAh
  
- Sumergible a 40 metros

Esta cámara es una de las más conocidas a nivel mundial dentro del campo de cámaras de acción (*GoPro* fue la primera empresa en sacar al mercado este tipo de cámaras) por su calidad de grabación y por disponer de una gran cantidad de accesorios para la práctica deportiva. Entre ellos se encuentran aquellos para hacer la cámara estanca.



**Figura 11:** GoPro Hero3+ Black (a) sin carcasa (b) con carcasa estanca

Para llevar a cabo la medición del pez se va a hacer uso de la visión estéreo, con lo que se necesitarán dos cámaras de características idénticas, que puedan sincronizar sus capturas y que sean sumergibles. *GoPro* nos dota de todas estas características mediante una carcasa dúo (separación entre lentes de 3,5 cm) estanca (Figura 12a) y un componente de sincronización (Figura 12b) para que las capturas de ambas cámaras coincidan en el tiempo.



**Figura 12:** (a) carcasa estanca para visión estéreo (b) elemento de sincronización de las cámaras (en naranja)

## 2. Calibración

La base de la visión estéreo es que, mediante la utilización de dos imágenes, se puede calcular la profundidad de cualquier punto del espacio en base a la diferencia de posición de su proyección en esas dos imágenes (Figura 13).

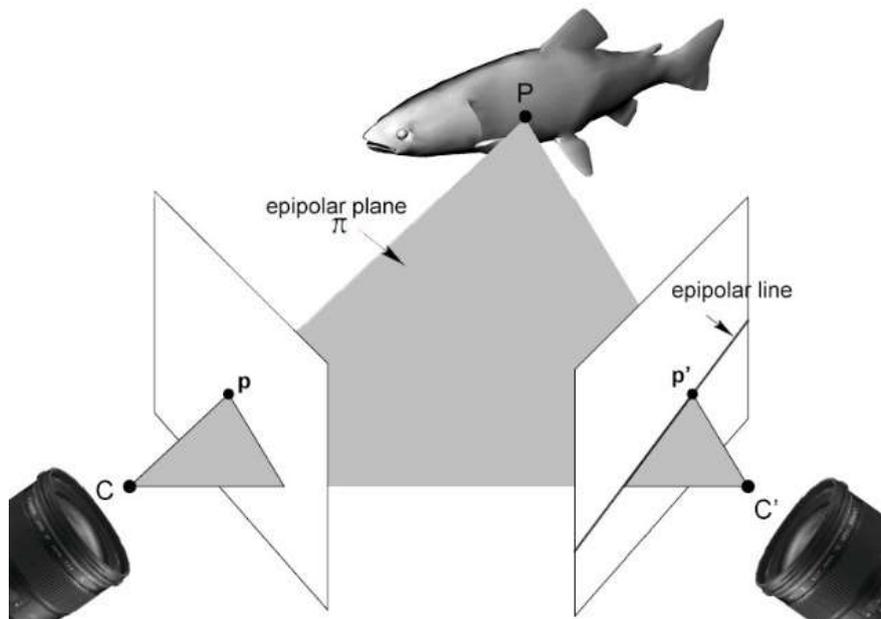


Figura 13: Geometría epipolar

Por ello, para llevar a cabo la técnica, se sincronizan dos cámaras con las mismas características, en este caso *GoPro Hero3+ Black*, de tal forma que se obtengan los dos frames simultáneos en cada instante de tiempo.

El principio de la estereoscopia puede ser explicada considerando un punto  $P$  en el espacio real (tridimensional), siendo proyectado simultáneamente en dos punto de dos imágenes ( $p$  y  $p'$ ) a través del centro óptico de las dos cámaras ( $C$  y  $C'$ ). Los puntos  $P$ ,  $p$ ,  $p'$ ,  $C$  y  $C'$  caen en un plano denominado plano epipolar. El plano epipolar intersecta cada imagen en una línea que corresponde a la proyección a través de  $p$  y  $P$ , y  $p'$  y  $P$  y que se conoce como líneas epipolares. Esta proyección está descrita por la geometría epipolar [67].

El primer paso para conseguir una buena técnica de visión estereo se sustenta en la realización de una calibración precisa de las cámaras. Mediante la calibración se

determinan los parámetros intrínsecos (distancia focal, desplazamiento del centro de la imagen, coeficientes de distorsión) y extrínsecos (translación, rotación) de la cámara para la transformación entre puntos 2D de la imagen y su proyección a puntos 3D de la escena. Este proceso de proyección puede ser descrito precisamente usando el modelo matemático *Pin-Hole* [68, 69], como indica la ecuación (1)

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}_M \times \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (1)$$

Donde  $(x, y, w)$  representa un punto  $p$  en coordenadas homogéneas,  $f$  representa la longitud focal de la cámara (la distancia de la lente al sensor de la cámara) y  $c$  determina las coordenadas de la imagen donde un punto es proyectado a través del centro de la lente  $C$ .  $(X, Y, Z)$  representa las coordenadas del punto  $P$ , en el espacio tridimensional. Por lo tanto, la matriz  $M$  modela la proyección de un punto de la imagen, a través de un sistema de cámaras hasta el espacio 3D real, obteniendo una línea de proyección  $pP$ , como se muestra en la Figura 13, con coordenadas planas  $(X/Z, Y/Z)$ .

Además, hay que tener en cuenta que el modelo *Pin-Hole* describe la proyección de un punto en un sistema óptico perfecto. En la práctica, la posición real del punto difiere de la ideal debido a que las imperfecciones en las lentes de la cámara introducen ciertas distorsiones en la imagen. Estas distorsiones siguen unas fórmulas matemáticas descritas en las ecuaciones (2) que dependen de un número reducido de parámetros  $D$  [70].

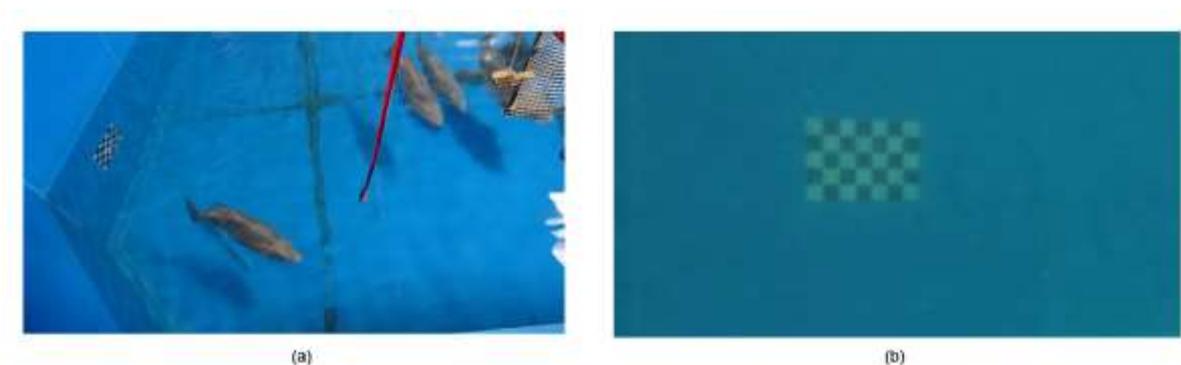
$$\begin{aligned} dr_x &= xk_1r^2 + xk_2r^4 \\ dr_y &= yk_1r^2 + yk_2r^4 \\ dt_x &= k_3(r^2 + 2x^2) + 2k_4xy \\ dt_y &= sk_3xy + k_4(r^2 + 2y^2) \\ D &= [k_1 \ k_2 \ k_3 \ k_4] \end{aligned} \quad (2)$$

Donde se modela una distorsión para un punto en la imagen, en las coordenadas  $(x,y)$ , siendo  $r = \sqrt{x^2 + y^2}$

El paso final para estimar la geometría epipolar del sistema de la cámara es la estimación de la traslación y rotación de la segunda cámara respecto a la primera (3).

$$\begin{bmatrix} C'_x \\ C'_y \\ C'_z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}_R \times \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}_T \quad (3)$$

Para el procedimiento de calibración de las cámaras (Figura 14a) y la obtención del modelo completo de la cámara, se requiere conocer una serie de puntos 3D y sus proyecciones en la imagen, estos se conocen como puntos de calibración. Por este motivo se utiliza una plantilla de puntos coplanarios (Figura 14b) y sobre él, el algoritmo de optimización *Levenberg-Marquardt*, minimizando de esta forma el error de proyección para todos los puntos de las vistas disponibles de ambas cámaras.



**Figura 14:** Proceso de calibración (a) con la plantilla de puntos coplanarios (b)

### 3. Transformación de la información capturada

En esta etapa, al venir las imágenes directamente de las cámaras, no hay que realizar ningún proceso de transformación ya que se genera directamente la matriz para trabajar con ellas.

#### 4. Pre-procesado

Debido a que las imágenes recogidas intentan recrear situaciones complejas del medio marino, como puede ser la turbidez, burbujas, cambios de luz y otras dificultades, que hacen que las imágenes tengan ruido o elementos ajenos a lo que se quiere detectar, se realiza previamente un pre-procesado de las imágenes buscando una mejor situación para la fase de procesado.

El primer paso consiste en convertir la imagen desde RGB a escala de grises, usando la fórmula (4).

$$\textit{Grayscale} = r.R + g.G + b.B \quad (4)$$

Para estos entornos, los valores por defecto ( $r=0.299$ ,  $g=0.587$  y  $b=0.114$ ) que proporciona OpenCV [51] son válidos. Sin embargo, en otros entornos naturales podría ser necesario realizar alguna modificación de los valores para obtener un mayor contraste en el frame, que facilite los siguientes pasos.

A continuación, se trata de reducir el ruido y mejorar la robustez de la técnica de substracción del fondo, con un filtro de desenfoque gaussiano. Este filtro usa una matriz gaussiana como máscara o matriz de convolución y produce valores de salida en cada pixel estableciendo un promedio ponderado de la vecindad de ese pixel. Como resultado de este proceso se suaviza la imagen y se conservan límites y bordes mejor que en otros filtros de desenfoque.

La ecuación (5) representa la función gaussiana en 2D que se aplica para este filtro, en donde  $\mu$  es la media y  $\sigma$  es la varianza para cada una de las variables  $x$  e  $y$ .

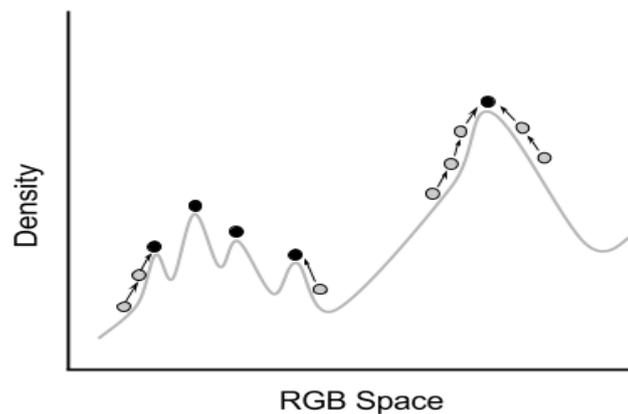
$$G_0(x, y) = A_e \frac{-(x - \mu_x)^2}{2\sigma_x^2} + \frac{-(y - \mu_y)^2}{2\sigma_y^2} \quad (5)$$

Como las imágenes de entrada tienen demasiado ruido y esto complica obtener buenos aciertos en la búsqueda de círculos con la transformada de Hough, se realizan ciertos pasos previos sobre las imágenes para poder llevar a cabo el proceso de búsqueda.

En el primer paso se hace uso del *Mean Shift Filter* (filtro de desplazamiento medio) [71, 72] con el objetivo de suavizar texturas y reducir el ruido, consiguiendo así una validación del color de los ojos más robusta y sencilla. Se dispone de una imagen RGB que consiste en  $n$  píxeles que contienen tres valores (r, g, b). Como una imagen es una distribución de puntos en el espacio tridimensional, una función de densidad puede ser definida como en la ecuación (6).

$$f(\vec{x}) = \{\vec{x}_i \mid i = 1, \dots, n\} \quad (6)$$

Los puntos de la función de densidad que son máximos locales, pueden ser detectados por el análisis de desplazamiento medio como se muestra en la Figura 15



**Figura 15:** Mean Shift Analysis

Cualquier punto en el espacio tridimensional converge a un máximo local, suavizando así la imagen. Además, la información de la posición también se puede utilizar para que los objetos alejados no estén etiquetados con los mismos colores. Para lograr esto, el vector tridimensional (r, g, b) se puede expandir a un vector de 5 dimensiones (x, y, r, g, b) que incluye la información de su ubicación.

Por otro lado, dado que debe mejorarse el contraste de los bordes de una imagen, la transformada de Hough es más robusta. También se aplica una máscara de enfoque a las imágenes para conseguir este objetivo. Para conseguir mayor nitidez se utiliza una versión ligeramente borrosa de la imagen original (desenfoco gaussiano), que

posteriormente se resta de la original para detectar sus bordes, creando una máscara de enfoque. Después se aumenta el contraste, junto con esos bordes, usando la máscara mencionada anteriormente (7).

$$Out(x, y) = 1.5 \times Input(x, y) - 0.5 \times G(x, y) \quad (7)$$

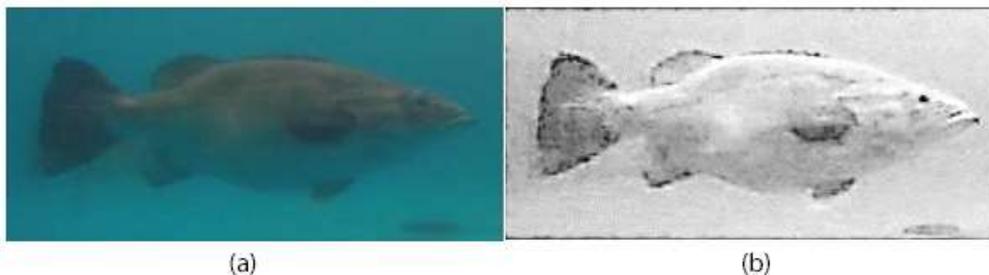
Debido a que este tipo de filtros pueden causar irregularidades en los bordes, que pueden dar lugar a bordes dobles, el Laplaciano de la imagen original también se añade a la imagen transformada (8).

$$Sharp(x, y) = Out(x, y) - W \times (Input(x, y) \times s \times L(x, y)) \quad (8)$$

En donde  $W$  es el peso,  $s$  la escala y  $L(x, y)$  el Laplaciano (9)

$$L(x, y) = \frac{\partial^2 Input}{\partial x^2} + \frac{\partial^2 Input}{\partial y^2} \quad (9)$$

En la Figura 16 se muestra un ejemplo de la imagen de entrada transformada a escala de grises y aplicados los filtros comentados.



**Figura 16:** Ejemplo de imagen filtrada (a) imagen Input (b) imagen Output

Con la imagen mejorada se inicia la eliminación del fondo para reducir los elementos de interés de la imagen. Ya que se está buscando un alto rendimiento que permita la detección en tiempo real del pez, la substracción de fondo que se implementa debe de ser lo más simple y rápido posible. Así lo primero que se aplica será un filtro de desenfoque gaussiano, para reducir el ruido de las imágenes de entrada.

A continuación, esta imagen resultante es comparada con una imagen del fondo que capta la cámara, usando para ello diferencias absolutas, empleando la ecuación (10).

$$AbsDiff(x, y) = |img_1(x, y) - img_2(x, y)| \quad (10)$$

Una vez computas las diferencias absolutas, se puede inferir qué píxeles pertenecen al fondo y cuales pertenecen a un objeto móvil. Sin embargo, incluso usando los filtros anteriores, el ruido puede causar problemas en forma de falsos positivos, con lo que se debe aplicar un umbral (11) que permita descartar la mayoría de estos resultados anómalos.

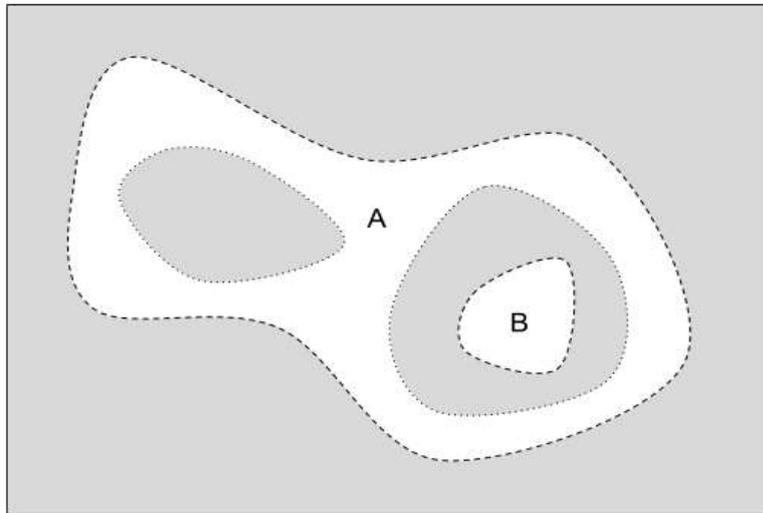
$$Thres(x, y) = \begin{cases} maxVal, & \text{if } img(x, y) > t \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

Después de descartar pequeñas diferencias como resultado de la aplicación del filtro previamente comentado, se van a expandir los aciertos (*matches*) encontrados, rellenando los pequeños huecos de los objetos encontrados. Para ello se utilizará otra técnica simple y rápida, que requiere poca capacidad de procesamiento, como la operación morfológica de dilatación (12). Se utiliza un rectángulo de 3x3 como elemento estructural.

$$Dilate(x, y) = \max_{(x', y') : element(x', y') \neq 0} Thresh(x + x', y + y') \quad (12)$$

Con el proceso de dilatación terminado, sobre la imagen resultante se aplica un algoritmo de detección de contornos [73]. Este algoritmo detecta los contornos de los objetos siguiendo los bordes en una imagen con píxeles frontera, que puede ser el resultado de, por ejemplo, un detector de bordes Canny [74]. De esta forma se genera una imagen binaria cuyo resultado son los bordes detectados. Este algoritmo diferencia entre contornos interiores y contornos exteriores con cero regiones (huecos).

En la Figura 17 se muestran contornos de los dos tipos mencionados, los contornos exteriores representados por líneas discontinuas y los contornos interiores por líneas de puntos.



**Figura 17:** Contornos exteriores e interiores en una imagen

Una vez se tienen calculados los contornos, se puede obtener de una manera sencilla el rectángulo delimitador mínimo del objeto detectado, generando así la región de interés del objeto (ROI). Dentro de esta ROI no se garantiza que el objeto detectado sea un pez, para validar que es un pez se va a aplicar la técnica que buscará un ojo de pez dentro de esa ROI.

## 5. Recepción de información en tiempo real

La información ya se está recogiendo en continuo mediante los vídeos sincronizados de las cámaras.

## 6. Detección de objetos

### *Detección de ojo por Hough*

Como se mencionó anteriormente, una vez se obtiene la ROI con un candidato potencial a ser pez, se requiere una validación que confirme si realmente ese objeto es un

pez. El método que se va a utilizar en esta sección trata de identificar el ojo del pez de forma que se permita realizar una diferenciación entre peces y ruido, burbujas, vegetación u otros tipos de seres vivos que se encuentran en el mundo marino. Es un método determinístico, es decir, no necesita entrenamiento (como por ejemplo las redes de neuronas artificiales), ya que se basa principalmente en la detección de círculos mediante la transformada de Hough.

Los primeros pasos que se realizan para tratar de detectar el ojo del pez, van orientados a mejorar la imagen de entrada, tratando de facilitar el trabajo a la transformada de Hough. Como se comentó en secciones anteriores, el filtrado (*Mean Shift Filtering*) se aplica para reducir el ruido y una máscara de enfoque para aumentar el contraste de los bordes.

Debido a los problemas que tiene la transformada de Hough con imágenes ruidosas y de bajo contraste, se utiliza un extractor MSER (*Maximally Stable Extremal Regions*) [75] para detectar objetos binarios grandes (*BLOB*), que muestren un grupo de píxeles conectados. Utilizando esta técnica se consigue reducir el número de falsos positivos de la transformada de Hough, que procesará la imagen binaria resultante, con solo los BLOBs blancos detectados sobre un fondo negro. MSER se podría utilizar también para detectar los círculos pertenecientes a los ojos de los peces, pero sería necesario medir los os parámetros del ojo para detectar candidatos.

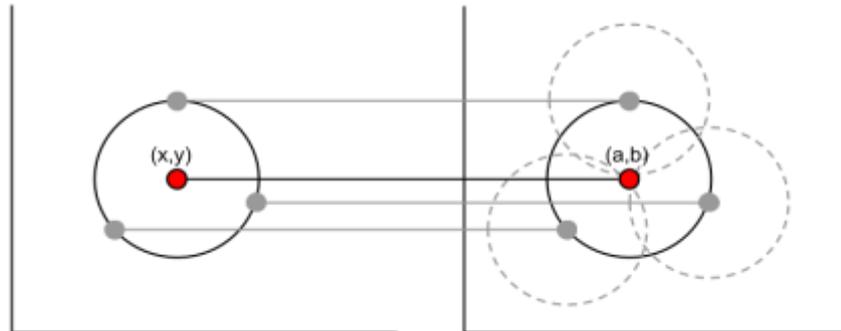
La transformada de Hough se utilizó inicialmente para detectar líneas, pero posteriormente se adaptó para otro tipo de figuras, como los círculos. Así la transformada de Hough para círculos [76, 77] puede ser utilizado para determinar los parámetros de un círculo cuando un número de puntos dentro de su perímetro son conocidos. Un círculo con radio  $R$  y centro  $(a,b)$  puede ser descrito con las paramétricas (13) y (14)

$$x = a + R \cos \theta \quad (13)$$

$$y = b + R \sin \theta \quad (14)$$

En el intervalo  $[0, 360]$  grados, los puntos  $x$  e  $y$  trazan el perímetro de un círculo (Figura 18). Este es un espacio paramétrico en 3D, en el que los parámetros del círculo

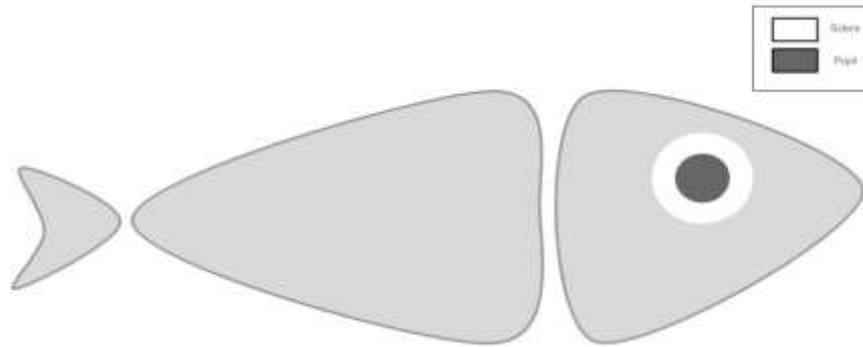
se pueden identificar por la intersección de muchas superficies cónicas definidas por puntos en el círculo en 2 dimensiones. El punto central real será común a todos los círculos de parámetros y se puede encontrar con una matriz de acumulados de Hough.



**Figura 18:** Cada punto del espacio geométrico (izquierda) es el centro de un círculo en el espacio paramétrico (derecha). Los círculos de dicho espacio se intersecan en  $(a, b)$  que corresponde al centro  $(x, y)$  en el espacio geométrico.

Como no se conoce el radio, el lugar del epicentro de los puntos cae sobre la superficie de un cono debido al cambio de radio  $R$ . En lugar de un círculo, cada punto en el perímetro del espacio geométrico del círculo produce una superficie de cono, por lo tanto, el vector  $(a, b, R)$  corresponde a la celda de acumulación donde se cruzan la mayoría de las superficies de los conos. Después de obtener los círculos candidatos, aún se necesita un procesamiento adicional para asegurar que ese candidato sea un ojo de pez. También se debe seleccionar el mejor candidato en caso de que se encuentren múltiples coincidencias ya que solo deberá aparecer un ojo por pez (objeto candidato).

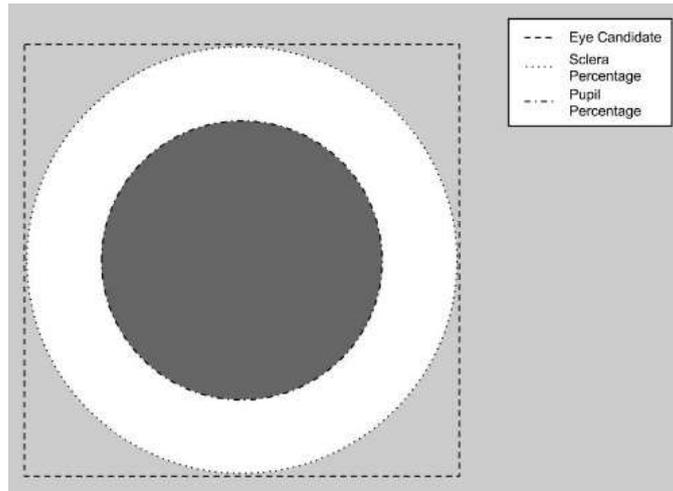
Con el fin de determinar si los círculos detectados son ojos de pez o no, se observaron los ojos de los peces que se iban a estudiar para desarrollar un sistema de puntuación para clasificar los círculos seleccionados como candidatos. Los tipos de peces que se van a intentar detectar son peces comunes que se parecen a la forma del patrón que se muestra en la Figura 19, en donde se aprecia las características de los ojos.



**Figura 19:** características utilizadas: la esclerótica y la pupila del ojo de pez.

Generalmente, los ojos de pez tienen una esclera o esclerótica [78], que es el blanco del ojo (Figura 19), y que es una capa exterior blanca, opaca y fibrosa que va desde la córnea hasta el nervio óptico y envuelve la pupila del ojo del pez. La pupila [79] es un orificio ubicado en el centro del ojo que permite que la luz incida en la retina. Su apariencia es de color negro debido a que los rayos de luz que entran en ella son absorbidos directamente por los tejidos interiores del ojo o absorbidos después de difundir los reflejos dentro del ojo, lo que provoca que no salgan de los límites de la pupila. Al contrario que en los humanos, en donde la lente es similar a una cámara plana que se encuentra debajo del iris y la pupila, la lente esférica del ojo de pez sobresale del iris a través de la apertura de la pupila. Por ello, el iris humano es ajustable según la intensidad de la luz, mientras que el iris de ojo de pez no lo es.

Estas características hacen del ojo de pez un rasgo característico para la diferenciación de los peces de otros objetos. El sistema de puntuación propuesto tiene en cuenta estos dos factores, la esclerótica y la pupila del ojo del pez. El sistema comprueba la diferencia entre los promedios de color en ambas regiones, ya que suele haber un alto contraste entre ellos como resultado de los colores blanco y negro. Para obtener las puntuaciones de los candidatos detectados y, teniendo en cuenta que diferentes peces pueden tener diferentes características oculares, el usuario debe establecer varios parámetros como el porcentaje de pupila, el porcentaje de esclerótica, el color objetivo de la pupila y el color objetivo de la esclerótica. La Figura 20 muestra lo que representan los parámetros.



**Figura 20:** Líneas que delimitan la esclerótica y la pupila del candidato a ojo

El porcentaje de pupila es el porcentaje de ojo que pertenece a la pupila, el resto del ojo será el porcentaje de esclerótica. Los dos parámetros restantes, el color objetivo de pupila y el color objetivo de la esclerótica, representan el valor promedio del color de la pupila y la esclerótica, respectivamente. Una vez elegidos los parámetros según el tipo de pez, el siguiente paso consiste en iterar sobre todos los píxeles del candidato y calcular los promedios de color de las diferentes características del ojo del pez. Suponiendo que la pupila y la esclerótica se pueden modelar con elipses, se debe comprobar si un determinado píxel está encerrado en la elipse definida por los porcentajes de pupila y esclerótica. Para ello se utiliza la fórmula (15)

$$\frac{(x - h)^2}{r_x^2} + \frac{(y - k)^2}{r_y^2} \leq 1 \quad (15)$$

Siendo  $x$  e  $y$  la posición del píxel,  $h$  y  $k$  la posición del centro de la elipse,  $r_x^2$  el semieje mayor y  $r_y^2$  el semieje menor. Si la desigualdad se cumple, entonces el píxel estará dentro de la elipse, en caso contrario caerá fuera de ella. Saber si un píxel cae dentro o fuera de pupila o de la esclerótica es sencillo: si el píxel está dentro de las dos elipses pertenecerá a la pupila; por el contrario, si está dentro de la elipse exterior pertenecerá a la esclerótica.

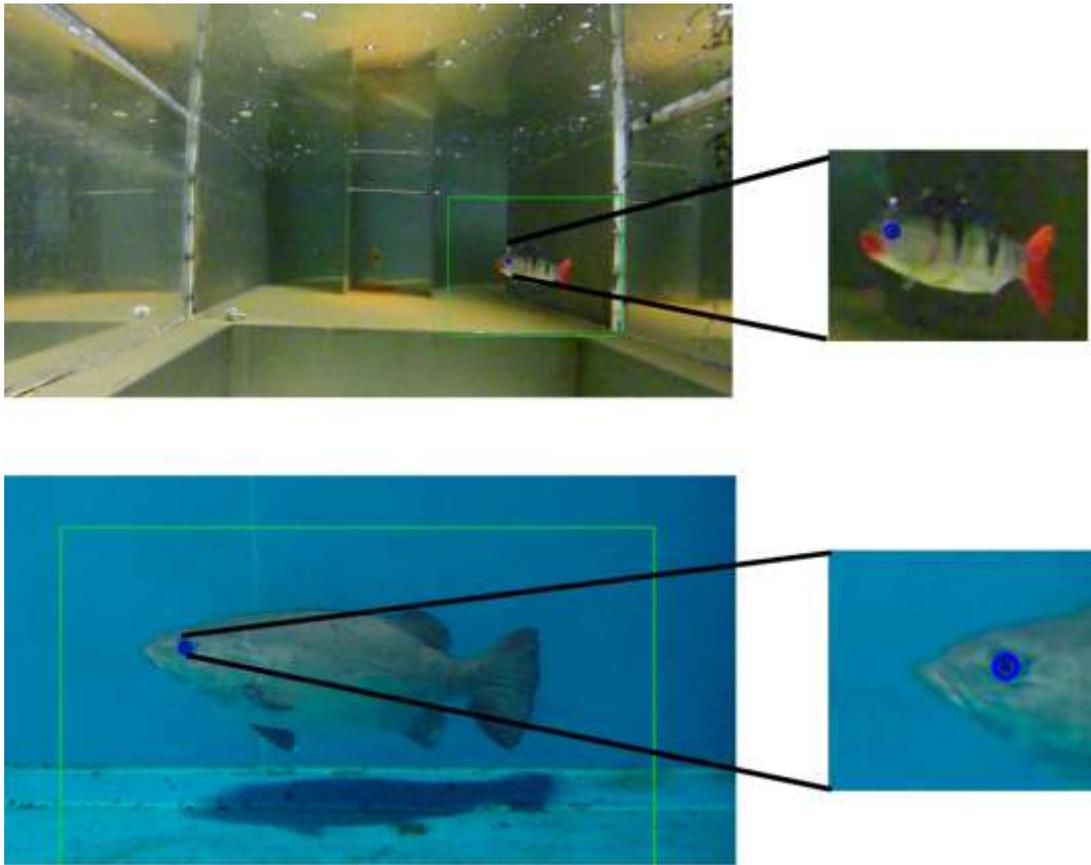
Como los píxeles se clasifican como parte de la esclerótica o la pupila, su color contribuye al promedio de color de una o de otra, que se calcula de la siguiente manera con las ecuaciones (16) y (17)

$$\bar{C}_s = \frac{1}{n_s} \sum_{i=1}^{n_s} img(x_s, y_s) \quad (16)$$

$$\bar{C}_p = \frac{1}{n_p} \sum_{i=1}^{n_p} img(x_p, y_p) \quad (17)$$

En donde  $n_s$  y  $n_p$  son el número de píxeles en la esclerótica y en la pupila,  $img$  es la imagen de entrada y  $x_s, y_s, x_p, y_p$  las posiciones de la esclerótica y la pupila en la imagen de entrada. Una vez calculado el promedio, se puede comparar con el color objetivo de la pupila y la esclerótica y se obtienen las diferencias entre los promedios calculados y el objetivo deseado. Si el error promedio supera un cierto umbral, la coincidencia se descarta, si no, se mantiene temporalmente. Para encontrar la mejor coincidencia, los candidatos se clasifican según su error y el que tiene el menor error se elige como la mejor coincidencia.

La Figura 21 muestra un ejemplo de un ojo de pez correctamente emparejado en una imagen submarina con ruido. En caso de que no haya una coincidencia de ojo, se puede suponer que el objeto detectado no es un pez o que no está en una posición adecuada para su medición.



**Figura 21:** Ejemplo de un ojo de pez (y zoom) correctamente emparejados que está resaltado en azul. El ROI en el que se ubicó el pez está marcado con un rectángulo verde.

La detección del ojo mediante la técnica de Hough tiene una tasa de precisión del 50% para peces de 9 centímetros, donde el pequeño tamaño del ojo hace que sea muy difícil de localizar, mientras que para peces de aproximadamente un metro, la tasa de precisión está entre el 60% y el 65%. El tiempo de procesamiento es inferior a un segundo, lo que permite utilizarlo en tiempo real procesando al menos una imagen por segundo (como se puede ver más adelante en la sección de resultados). Sin embargo, el ajuste del 60% no alcanza el objetivo final, por lo tanto, es necesario implementar otra técnica que permita mejorar esta capacidad de detección. Cuando esta técnica no es capaz de detectar al pez se va a utilizar *Deep Learning* para su detección mediante una red de neuronas artificiales que clasifique si hay pez o no. Esta técnica es mucho más costosa computacionalmente hablando, por lo que solo se aplicará cuando falle el algoritmo anterior.

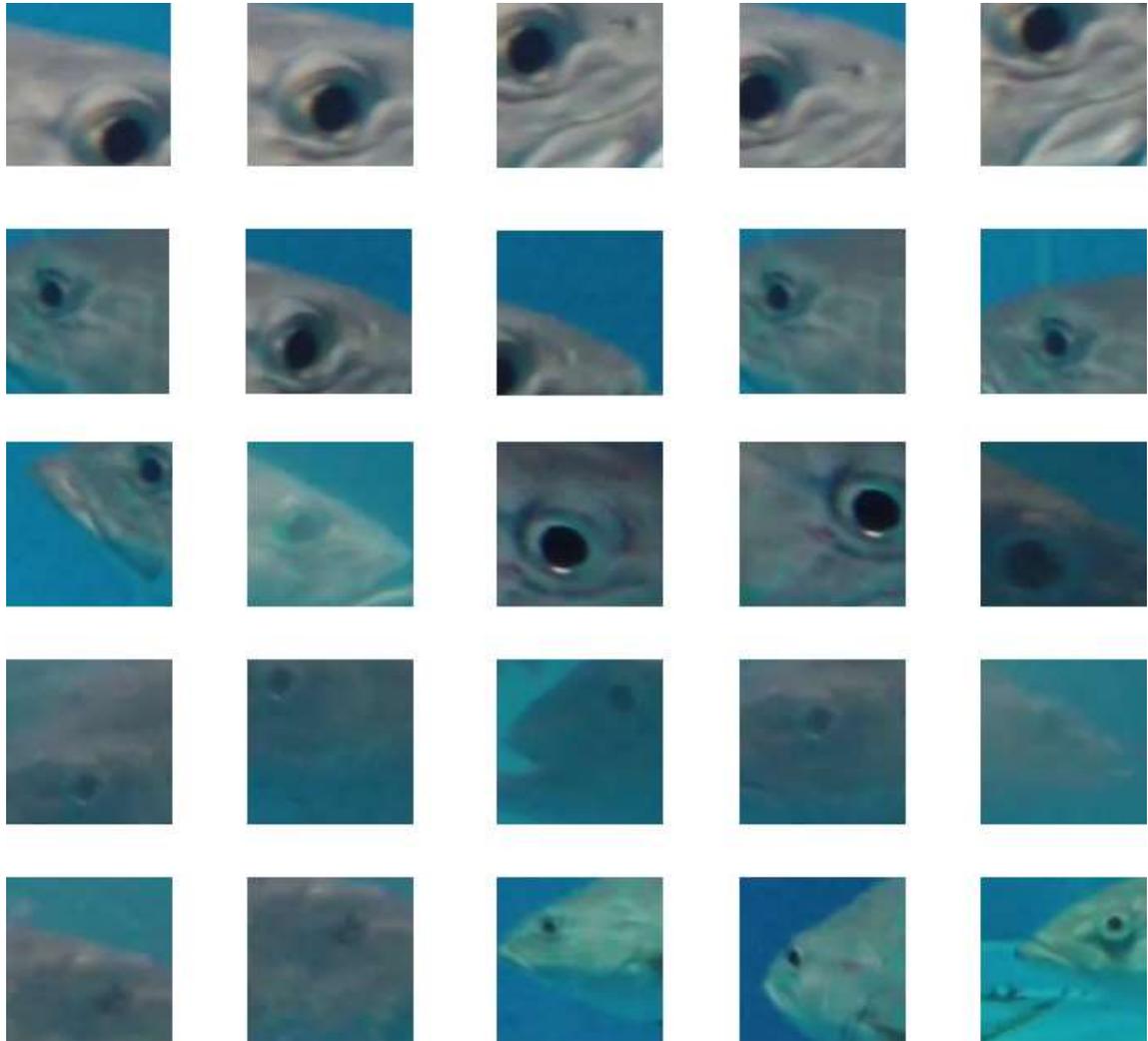
### *Detección usando FF RNA*

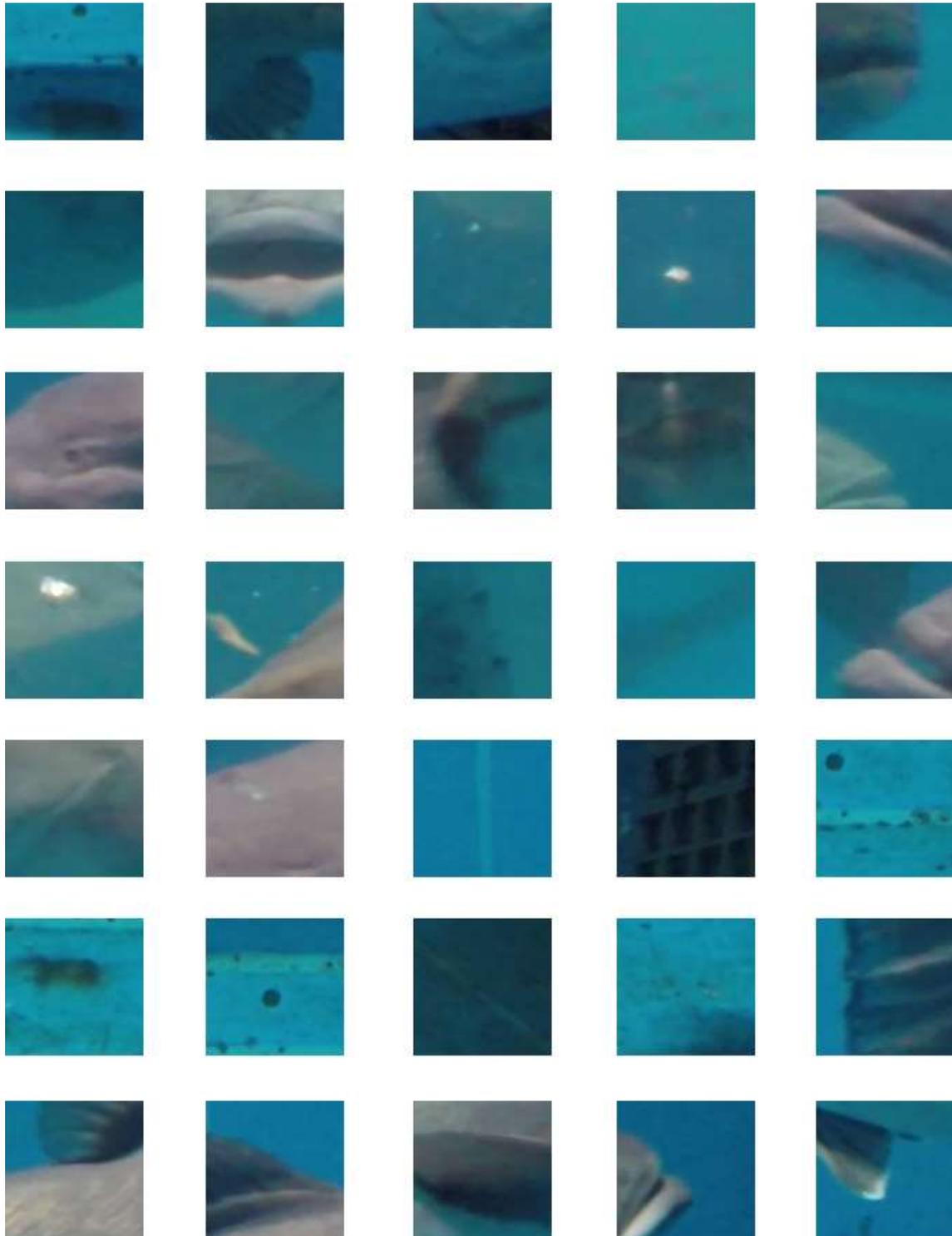
Se utiliza la técnica de *Deep Learning* mediante una Red de Neuronas Artificiales (RNA) con Feed-Forward [80] para enfocar como un problema de clasificación la detección del pez. La salida de la RNA deberá indicar si hay algún ojo de pez en la imagen.

La estructura de una RNA Feed-Forward está formada por una capa de entrada, una capa de salida y una capa intermedia que es una multicapa de capas ocultas. En este trabajo, la capa intermedia tendrá una única capa para reducir así el tiempo de ejecución.

El uso de RNA requiere un proceso de entrenamiento como paso inicial. En el proceso que se sigue en este trabajo, habrá un conjunto de entrenamiento formado por imágenes clasificadas como “ojo” o “sin ojo”. Todas estas muestras se recogen de imágenes completas de peces dentro de tanques, recortando la muestra al tamaño deseado. En este caso, se van a utilizar imágenes de 100 píxeles de ancho y 100 píxeles de alto para evitar tiempos de entrenamiento demasiados altos (con esta configuración el proceso de entrenamiento de la RNA requiere entre 3 y 4 horas).

En la Figura 22 se ilustra una muestra del conjunto de entrenamiento. El conjunto completo de entrenamiento está formado por 69 imágenes de ojos y 103 imágenes que no pertenecen a ojos. Estas imágenes recortadas se utilizan como muestras de entrada a la red, por lo que la red tendrá un elemento de entrada para cada píxel de la imagen (10000 elementos de procesado). Del conjunto total de entrenamiento, el 85% (146 imágenes) se utilizan para entrenar la red y el restante (26 imágenes) se reservan para la fase de validación.





**Figura 22:** Extracto del conjunto de datos utilizado para el entrenamiento y la validación de RNA

Se realizan varios ensayos con diferentes arquitecturas y parámetros de configuración. Se prueban arquitecturas con 500, 1000, 2000, 5000 y 7500 neuronas en la capa oculta y se observa que los mejores resultados se consiguen utilizando 2000

neuronas en la capa oculta y haciendo 7500 ciclos (*epochs*: número de veces que los datos de entrenamiento se pasan por la red para entrenarla).

Con esta configuración la RNA pudo predecir 21 de las 26 imágenes de validación (80%) y 145 de 146 imágenes de entrenamiento (99,31%).

Con esta técnica se consiguen buenos resultados, aunque se detectan demasiados falsos positivos, por lo que sería necesaria una fase de post-procesamiento. Además, el principal inconveniente de este proceso es el tiempo requerido, ya que implica el recorte de la imagen original en segmentos de 100x100 píxeles y la ejecución de una RNA de dimensiones considerables con cada uno de ellos, lo que implica un coste considerable en cuanto a tiempo de ejecución y requisitos computacionales (más de 10 segundos). Esta es la razón principal por la que se utiliza solo la RNA una vez que el algoritmo de Hough no detecta los peces.

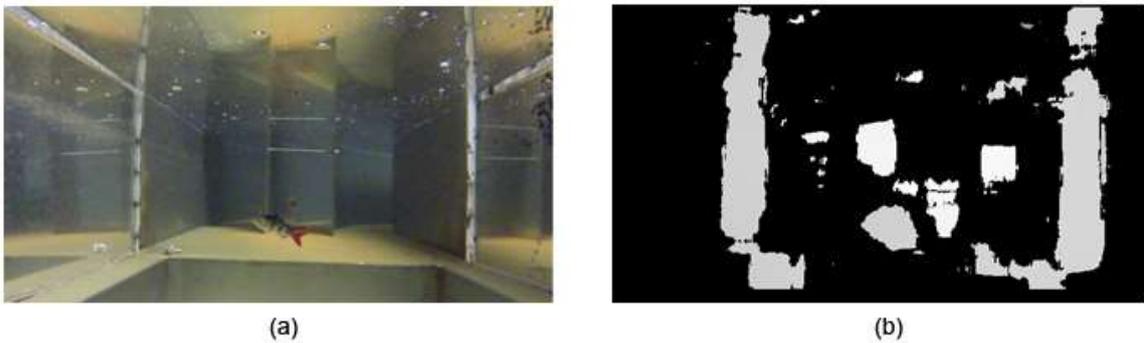
## 7. Seguimiento de objetos

Para el seguimiento del pez se va a utilizar la imagen rectificadas sin el fondo, que se verá en el punto siguiente, y se realiza el pronóstico de la posición del punto medio del rectángulo en el que está el contorno del pez, combinado con otras propiedades como el área del cuadrado y dimensiones. También se controla de una manera automática que solo se realicen mediciones cuando el pez aparece entero en el frame, eliminando falsas mediciones en los momentos de entrada y salida de la imagen.

## 8. Obtención del tamaño del pez

Con los peces ya detectados se procede a realizar una estimación de su tamaño, con lo que se necesita posicionar el pez en imagen en 2D en una escena tridimensional. Para estimar un mapa 3D de una escena, una vez se ha obtenido el modelo de las cámaras con la calibración realizada, se lleva a cabo un proceso de búsqueda de los puntos de correspondencia cuyo resultado es la estimación de un mapa de disparidad (Figura 23). La disparidad está inversamente relacionada con la distancia a la que se encuentra el objeto y su mapa muestra las diferencias en el posicionamiento de los puntos de correspondencia de las dos cámaras.

Para calcular este mapa de disparidad se ha utilizado el algoritmo de *Block-Matching* disponible en *OpenCV*. La imagen es dividida en varias partes que son asignadas a un bloque para calcular su mapa de disparidad. Cuando todos los bloques hayan terminado se forma el mapa de disparidad completo. Esta técnica analiza la similitud estadística de los valores de los píxeles en cada región (o bloque) de la imagen. El propósito es resolver el problema de correspondencia para cada bloque, encontrando, en la siguiente imagen, la región que representa el desplazamiento más probable.



**Figura 23:** (a) Imagen real (b) mapa de disparidad de la imagen

Numéricamente, el punto  $(i', j')$  se corresponde con el punto  $(i, j)$  después de aplicar el desplazamiento  $d(i, j)=(dx, dy)$ , que se puede describir como indica la ecuación (18).

$$(i', j') = (i + x, j + y) \quad (18)$$

Esta aproximación se basa en el supuesto expresado en (19)

$$I(i, j) + N = I'(i + x, j + y) \quad (19)$$

Siendo N un factor de ruido que sigue una distribución estadística específica, I la imagen obtenida de la cámara C e I' la obtenida de la cámara C'.

En este punto, una segmentación de la imagen basada en un umbral de la disparidad, servirá para obtener los objetos ubicados en el campo de visión de las cámaras y localizados a una distancia de interés.

### *Medición*

Tras realizar la calibración de la cámara estereo, y obtenidos los modelos de la cámara, y la posterior obtención del mapa de disparidad se puede realizar el proceso de estimación del pez. En este proceso se van a utilizar la matriz de reproyección Q (23), parámetro calculado durante la calibración, que relaciona las coordenadas de la imagen en el escenario 2D con el 3D.

$$Q = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & -1/T_x & (c_x - c'_x)/T_x \end{bmatrix} \quad (20)$$

Donde  $c_x$  y  $c_y$  son las coordenadas del punto principal de la cámara izquierda,  $f$  es la distancia focal de la cámara izquierda,  $T_x$  es la translación en el eje x y  $c'_x$  es la coordenada x del punto principal de la cámara de la derecha.

Sobre la imagen con el fondo eliminado se aplica una remapeo, según los parámetros de calibración de la cámara, de tal forma que la imagen queda rectificadas (eliminando distorsiones) pudiendo así relacionarla de una forma más precisa con el mapa

de disparidad obtenido ya que de otra forma podrían producirse contradicciones entre las coordenadas de esta imagen y los datos recabados en el mapa de disparidad. A continuación, se extraen los contornos de los peces detectados anteriormente y el mínimo rectángulo que los contiene, lo que permite conocer las coordenadas en las que se encuentran los objetos en la imagen 2D, o su área y otros parámetros descriptores de los objetos que permitan realizar un filtro de objetos que se desean encontrar.

En esas coordenadas se busca el contorno del mapa de disparidad correspondiente a ese objeto y se calcula la distancia a la que se encuentra el objeto, lo que permitirá eliminar objetos que se encuentren a distancias más cercanas o lejanas de lo que se desee. Para estimar la distancia se toma el punto medio del contorno y se calcula, utilizando la matriz  $Q$ , según la ecuación (21)

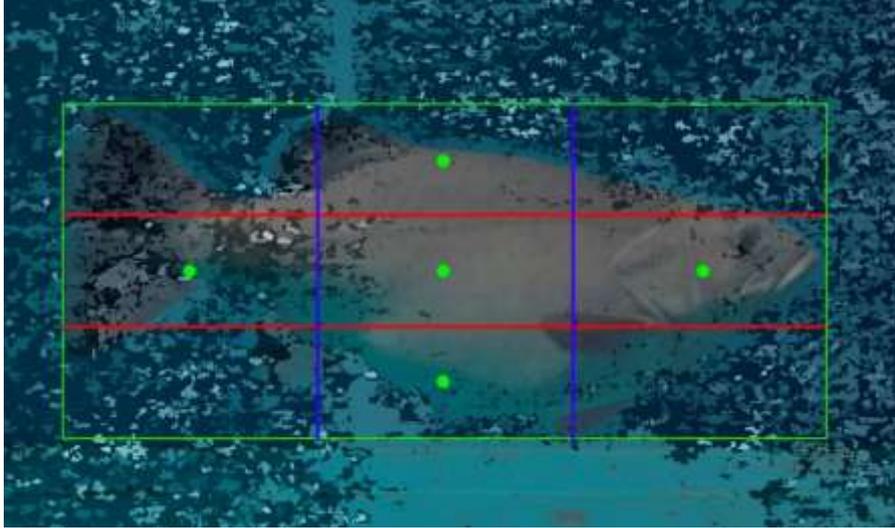
$$Q \times \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} \quad (21)$$

En donde  $x$  e  $y$  son las coordenadas del punto y  $d$  es la disparidad en ese punto. De la ecuación (21) se concluye que las coordenadas 3D del punto serán (22):

$$P = (X/W, Y/W, Z/W) \quad (22)$$

Para los objetos, dentro del rango de distancias que se desea, se va a establecer una malla de puntos de interés que permitan realizar una estimación más precisa de su tamaño. Esto se realiza debido a que el objeto puede estar en una posición más o menos angulada o no completamente paralelo al eje de la cámara estéreo, con lo que su distancia a ella puede ser diferente en cada punto. Horizontalmente, se divide el objeto en tres bloques del mismo tamaño y se busca el punto medio de cada uno para calcular su disparidad (Figura 24). En caso de que la disparidad en ese punto sea menor que la mínima disparidad posible o no haya sido posible calcular la disparidad, se busca verticalmente en ese punto el primer valor correcto de la disparidad y que esté dentro del contorno. En la otra dimensión se procede de la misma manera, realizando la búsqueda horizontalmente si es necesario encontrar otro punto con la disparidad correcta. Además se establece un

criterio en el que cuando la variación entre disparidades de los puntos sea muy grande, lo que indicaría que el objeto ya no está en una posición idónea para ser medido, no se realicen estimaciones erróneas que reducirían la precisión del algoritmo.



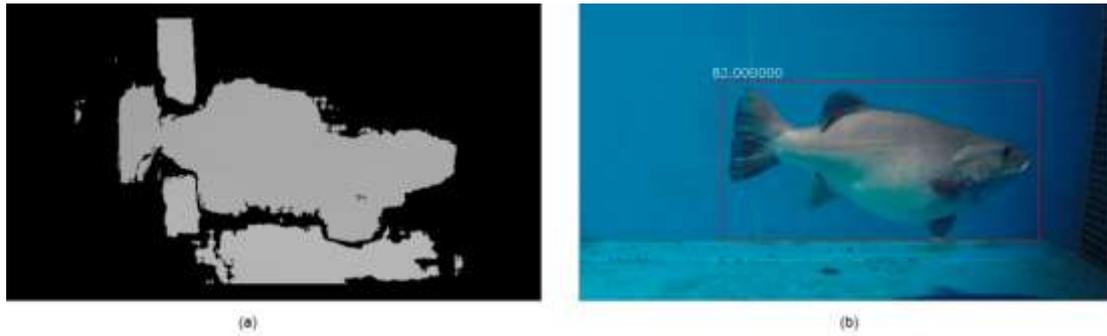
**Figura 24:** División de la ROI para estimar el tamaño

Una vez se tienen las coordenadas en la imagen y la disparidad de cada bloque, se procede a calcular la proyección en el mundo 3D de X e Y para posteriormente realizar la suma de las mediciones de las distancias de cada bloque (23).

$$\text{sizeEstimationX} = \sum_{i,j=0}^2 \left( \text{abs} \left( \frac{x_i \times Q[0,0] + Q[0,3]}{d \times Q[3,2] + Q[3,3]} - \frac{x_j \times Q[0,0] + Q[0,3]}{d \times Q[3,2] + Q[3,3]} \right) \right) \quad (23)$$

$$\text{sizeEstimationY} = \sum_{i,j=0}^2 \left( \text{abs} \left( \frac{y_i \times Q[1,0] + Q[1,3]}{d \times Q[3,2] + Q[3,3]} - \frac{y_j \times Q[1,1] + Q[1,3]}{d \times Q[3,2] + Q[3,3]} \right) \right)$$

La medición resultante se sitúa en un buffer de medidas del pez, que existe mientras el objeto aparece en la imagen, y se calcula la media de los valores del buffer para estimar el valor final del tamaño (Figura 25).



**Figura 25:** (a) Mapa de disparidad para la detección (b) estimación del tamaño del pez (longitud obtenida 82cm; longitud real 80.9 cm)

## 9. Visualización en el ordenador

Para la visualización se ha desarrollado un aplicación de consola sencilla, a modo de prototipo, en donde se muestra una ventana para reproducir lo que envía cada cámara. Otra ventana mostrará los posibles peces detectados y en otra se genera el mapa de disparidad para la zona de la imagen en la que está el pez detectado (Figura 25a). Por último se genera una imagen salida en la que se sobrepone la captura de una de las cámaras junto con el pez detectado encuadrado dentro de un rectángulo y sobre el cual se imprime la medición estimada (Figura 25b).

En un futuro esta herramienta se desarrollará en forma de software de escritorio, para permitir más interacción del usuario con la herramienta, permitiéndole modificar ciertos parámetros para hacer una detección más fina.

## 5.3. Resultados experimentales

Para medir la precisión del sistema propuesto para la estimación del tamaño del pez, se han llevada a cabo diferentes experimentos en diferentes entornos y condiciones. Para realizar estos ensayos se han utilizado vídeos realizados en laboratorio, donde el ambiente está más controlado, concretamente en las instalaciones del Centro de Innovación en Construcción e Ingeniería Civil (CITEEC), así como otros vídeos grabados

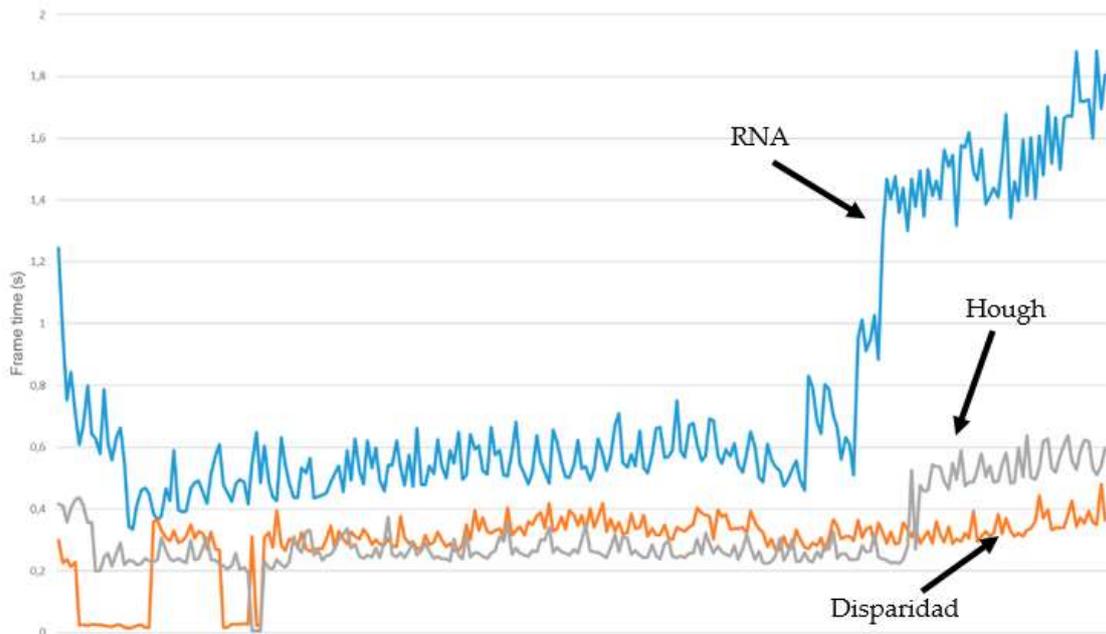
en entornos más similares a escenarios reales, como las instalaciones del Aquarium Finisterrae de A Coruña. En total se analizaron más de 3000 imágenes.

Las pruebas de rendimiento incluyen puntos de referencia que prueban la eficiencia del sistema en múltiples combinaciones con peces, sin peces, varios peces, diferentes posiciones de peces, diferentes combinaciones de turbidez y luminosidad, burbujas, etc., mientras que los puntos de referencia de robustez prueban la corrección del método propuesto en términos de detección y medición de peces. El sistema que se ha utilizado para probar el software desarrollado es un ordenador de rendimiento medio con el siguiente hardware:

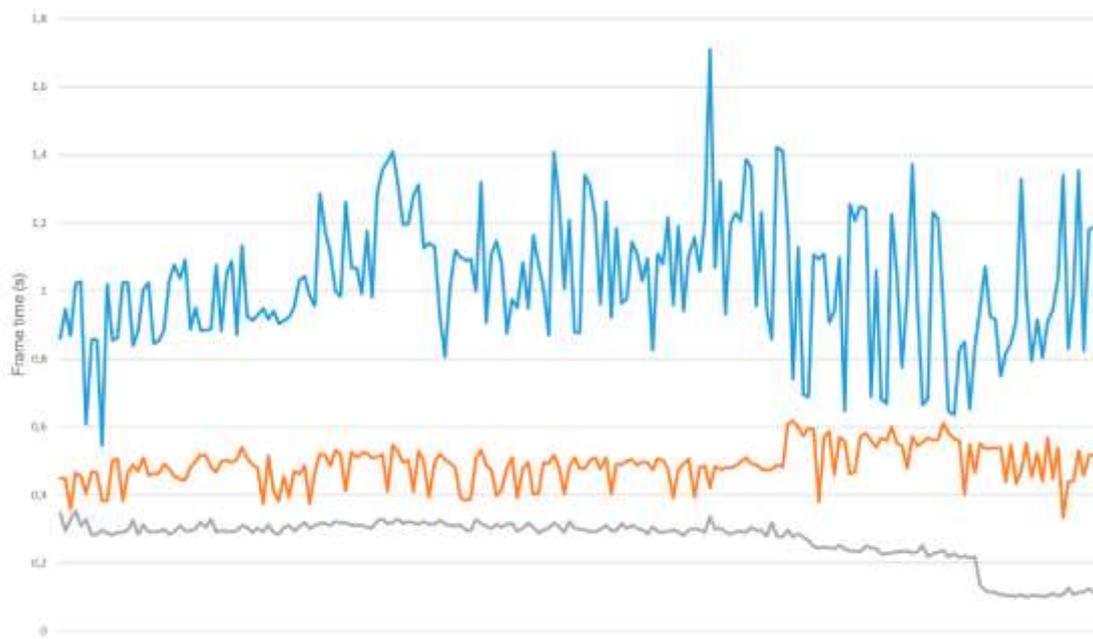
- Procesador: Intel Core i7-3770 CPU (4 cores, 8 threads)
- Tarjeta gráfica: NVIDIA GT 640
- RAM: 16 GB de 1600 MHz DDR3
- Disco duro: WDC WD10EZRX

El sistema operativo utilizado para ejecutar las pruebas fue Windows 7 en su versión de 64 bits, el compilador usado para generar el código fue VC++ v110 y la versión de *OpenCV* utilizada fue la 3.0

La ejecución de la técnica fue probada en tiempo real. Los gráficos, con los tiempos de ejecución de la técnica propuesta en dos vídeos, se muestran a continuación. La Figura 26 muestra el sistema probado con el denominado Vídeo 1. Esta escena contiene un solo pez de prueba de 9 cm de largo, que nada a diferentes profundidades. Además, este vídeo contiene una gran cantidad de ruido, burbujas y luz cambiante, lo que a su vez presenta un serio desafío para los detectores. El cálculo del mapa de disparidad de los peces detectados no es demasiado complejo ya que el pez tiene colores brillantes y varios rasgos identificativos, lo que permite que el algoritmo obtenga buenos resultados. Los resultados del rendimiento de la segunda prueba se puede ver en la Figura 27, donde el sistema se prueba con lo que se llama Vídeo 2. Este es un escenario más complejo, ya que hay varios peces de un metro de largo nadando en el tanque.

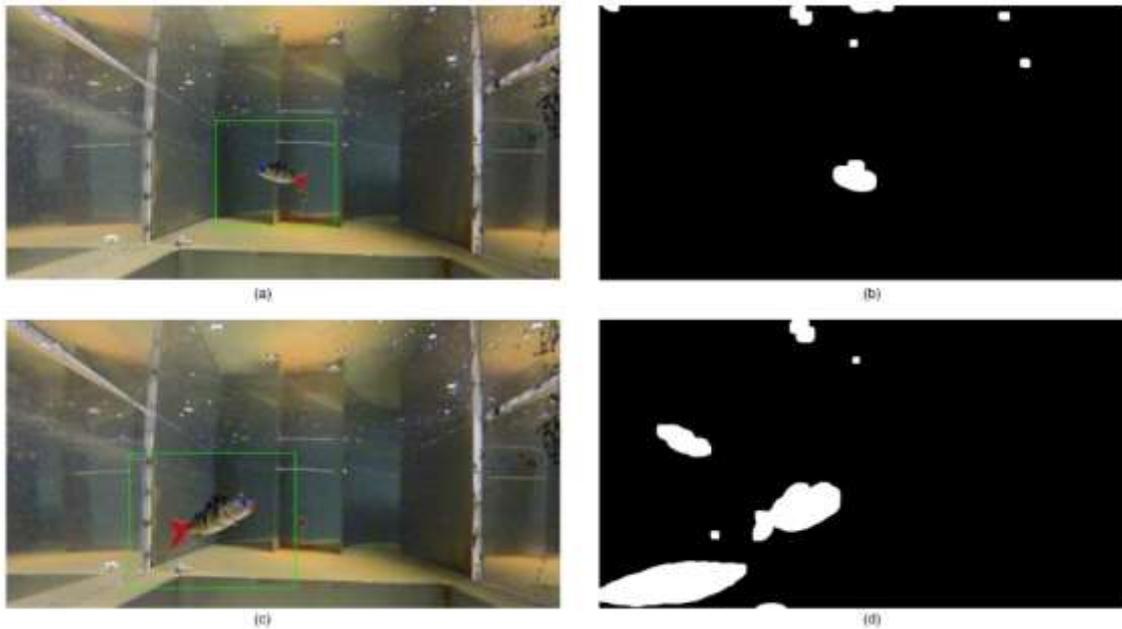


**Figura 26:** Ejecución de algoritmos en el Vídeo 1



**Figura 27:** Ejecución de algoritmos en el Vídeo 2

Los picos más altos de la Figura 26 se producen por cambios de luz y sombras cuando el pez está al lado de las paredes del tanque. Esto provoca cambios al realizar la substracción de fondo, como se muestra en la Figura 28, y como el algoritmo de Hough falla en la detección del pez, la RNA debe analizar las imágenes, que lleva más tiempo de procesamiento.



**Figura 28:** (a) y (c) detecciones del pez (b) y (d) se observan cómo afectan los cambios de luz y las sombras

Ambos gráficos muestran que, para lograr que el procesamiento pueda ser hecho en tiempo real, usar solo el algoritmo de Hough permite obtener un fotograma por segundo, pero si se quiere aumentar la precisión cuando este método no detecta los peces, y por lo tanto usar la RNA, los picos de tiempo de ejecución durante el procesamiento superan el segundo. Este incremento de tiempo viene ocasionado por el preprocesamiento que se debe de realizar para reducir el ruido, el impacto de la luz cambiante y la validación de los resultados proporcionados por ambos algoritmos (Hough y RNA). Al usar *Deep Learning* con la red entrenada, cuando falla el algoritmo de Hough, el sistema tiene una tasa de precisión del 74% con el conjunto de validación y casi el 100% con el conjunto de entrenamiento.

<i>Results</i>	<i>European perch</i>	<i>Brown trout</i>	<i>Atlantic wreckfish</i>
<i>Avg. Measured Size</i>	8.8	6.5	92.6
<i>Std. Dev. Measured Size</i>	0.8	0.8	8.4
<i>Avg. Absolute Error (cm)</i>	0.6	0.9	6.5
<i>Std. Dev. Absolute Error (cm)</i>	0.6	0.7	5.2
<i>Avg. Relative Error</i>	0.07	0.12	0.07
<i>Std. Dev. Relative Error</i>	0.09	0.09	0.06

<i>True Positives</i>	620	600	182
<i>Detected False Positives</i>	44	31	47
<i>Precision</i>	0.9	1.0	0.8
<i>False Positive Ratio</i>	0.1	0.0	0.2

**Tabla 1:** Resultados de ensayos

## 5.4. Conclusiones

En este caso práctico se ha mostrado una aplicación de técnicas de Inteligencia Artificial para la detección de peces. La ventaja del método propuesto en este estudio es que no es necesario tener un conocimiento profundo de las especies de peces para detectarlas. Sin embargo, al igual que otras técnicas de Visión Artificial, sigue estando condicionada a una buena calidad de imagen y una buena sustracción de fondo. Otras ventajas de esta técnica son que la detección se realiza en tiempo real y que la grabación se realiza en formato estéreo, y estos datos pueden utilizarse posteriormente para calcular el tamaño del pez.

El método de la transformada de Hough, incluso con el uso del MSER para aumentar la robustez, tiene dificultades con imágenes ruidosas y también depende en gran medida de las condiciones de luz. Por esta razón, también se ha proporcionado un método alternativo de detección de peces que consiste en la utilización del *Deep Learning* con el uso de una RNA. Al comparar el método de transformada de Hough con el uso de redes neuronales, este último detecta correctamente a los peces en el entorno para el cual ha sido entrenada pero, cuando las condiciones varían, la red debe reconstruirse y volver a entrenarse para este nuevo escenario.

Otros trabajos se basan en la forma o el color del pez o en el fondo para obtener una segmentación que permita su detección, pero esto hace que cuando cambie la especie o las condiciones el método pierda su utilidad. Por otro lado, la comparación de patrones tiene un gran inconveniente que consiste en la necesidad de tener suficientes imágenes de la especie a detectar. Por ello, la propuesta aquí detallada, podría realizar una búsqueda

entre los patrones disponibles, y utilizando ambos métodos (algoritmo de transformación de Hough y, cuando falle, la RNA) se podría obtener una precisión del 74%.

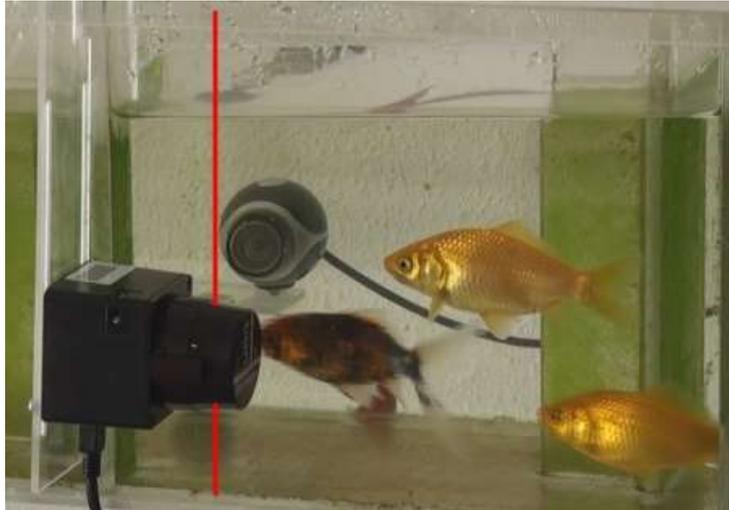
El primer paso para probar la técnica fue incluir más de una especie en el mismo ambiente para que varios especímenes de diferentes especies pudieran aparecer en el mismo *frame*. En segundo lugar, las pruebas se han realizado en un entorno lo más cercano posible a la realidad, donde la turbidez y los cambios de brillo tienen mayor impacto. Los resultados obtenidos han sido satisfactorios: en un vídeo real de 3000 segundos (50 minutos) de un banco de peces en un acuario, el método ha sido capaz de detectar el 74% de las imágenes en las que aparecía un pez. Esto significa que es posible realizar un sistema de detección de peces en tiempo real (una imagen por segundo) que sea capaz de detectar peces de manera eficiente y posteriormente detectar su tamaño. De esta forma podría aplicarse en casos prácticos como piscifactorías o en estudios de poblaciones silvestres, donde el objetivo es analizar la evolución del tamaño de un banco de peces.

## **6. Caso práctico 2: Detección de peces en escala de hendidura vertical combinando técnicas de visión artificial con la tecnología láser**

### **6.1. Introducción**

Para el desarrollo de esta técnica se van a combinar diferentes elementos. Por un lado están los sensores. Se dispondrá de una cámara destinada a obtener la imagen en el momento en que un objeto pase por la zona de influencia escaneada por el segundo sensor que forma parte de esta técnica. Este sensor es un sensor láser, concretamente el que se utiliza para los ensayos tiene una longitud de onda de 785 nm) que en cuanto un objeto se interpone en su haz, lo detecta y devuelve información sobre este objeto. Para la captura de datos, estos dos sensores se colocan enfrentados en las paredes laterales del canal, como se muestra en la Figura 29.

La información obtenida de los sensores se vuelca y se almacena en un ordenador en donde se ejecuta una aplicación software diseñada especialmente para esta técnica y en donde se aplicarán las técnicas de visión artificial necesarias. Además hay una unidad de control para que el usuario pueda interactuar con la aplicación.



**Figura 29:** Posición de los sensores en la técnica propuesta

La técnica propuesta tiene dos fases diferenciadas. En primer lugar hay una etapa en la que se van recogiendo las lecturas del sensor láser que envía en cada escaneo y con ellas se genera una imagen en tiempo real de la zona que cubre el sensor. Sobre ella se llevará a cabo un proceso de visión artificial con un pre-procesado, posteriormente una segmentación y se terminará con una fase de representación e interpretación.

Una vez terminado el proceso de visión artificial sobre la imagen del sensor láser se podrá determinar si se ha detectado algún objeto pasando por la zona de escaneo del láser. En ese caso, se iniciará la segunda fase de la técnica buscando, como objetivo, la verificación de si el objeto detectado es un pez. Si ha sido así, se inicia la segunda fase con una captura de la cámara que cubre la zona de influencia del láser y sobre esta imagen se realizará un nuevo proceso de visión artificial, repitiendo las etapas de la primera fase, esto es, pre-procesamiento, segmentación, representación e interpretación. Al término de este proceso el sistema deberá ser capaz de confirmar si un pez ha sido interceptado por el láser o no. En caso afirmativo, el sistema incrementará un contador que lleva implementado para conocer el número de peces detectados.

Con estos dispositivos se cumple uno de los requisitos deseados con esta técnica, desarrollar un sistema de detección de peces que sea capaz de realizar la tarea en tiempo real, con un coste computacional más ligero del que se pueda tener con, por ejemplo, redes de neuronas artificiales y que a mayores pueda ser llevado a la práctica con un coste bajo.

Así pues, el sistema completo queda formado por una parte software, que almacenará los datos que vaya capturando, que sea capaz de realizar las técnicas de visión artificial y que permita al usuario, mediante una unidad de control, comunicarse con el usuario. Y por otro lado una parte hardware que será la formada por los sensores, tanto la cámara como el láser.

## 6.2. Metodología

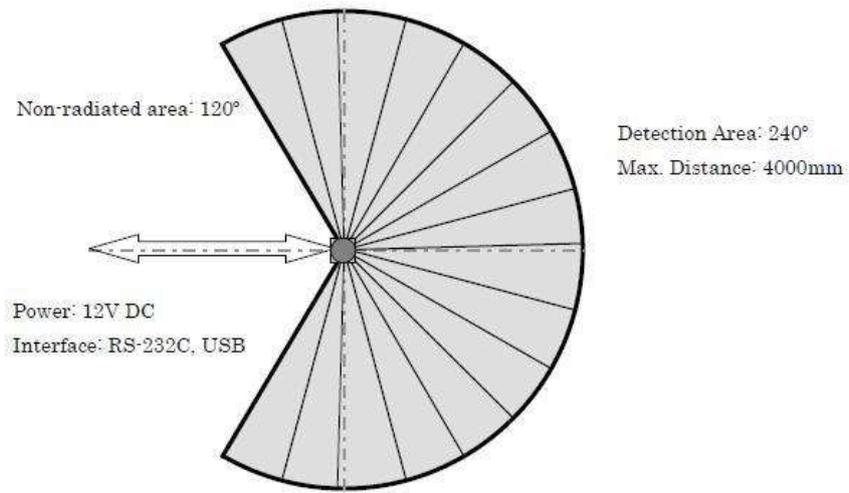
### 1. Dispositivos hardware de captura

Se utiliza el sensor láser URG-04LX-UG01 [81] del fabricante japonés *Hokuyo* (Figura 30). Es un sensor con fuente de luz láser infrarrojo y cuya longitud de onda es de 785nm. Su área de escaneo es un semicírculo de 240° con un radio máximo de 400mm (Figura 31).



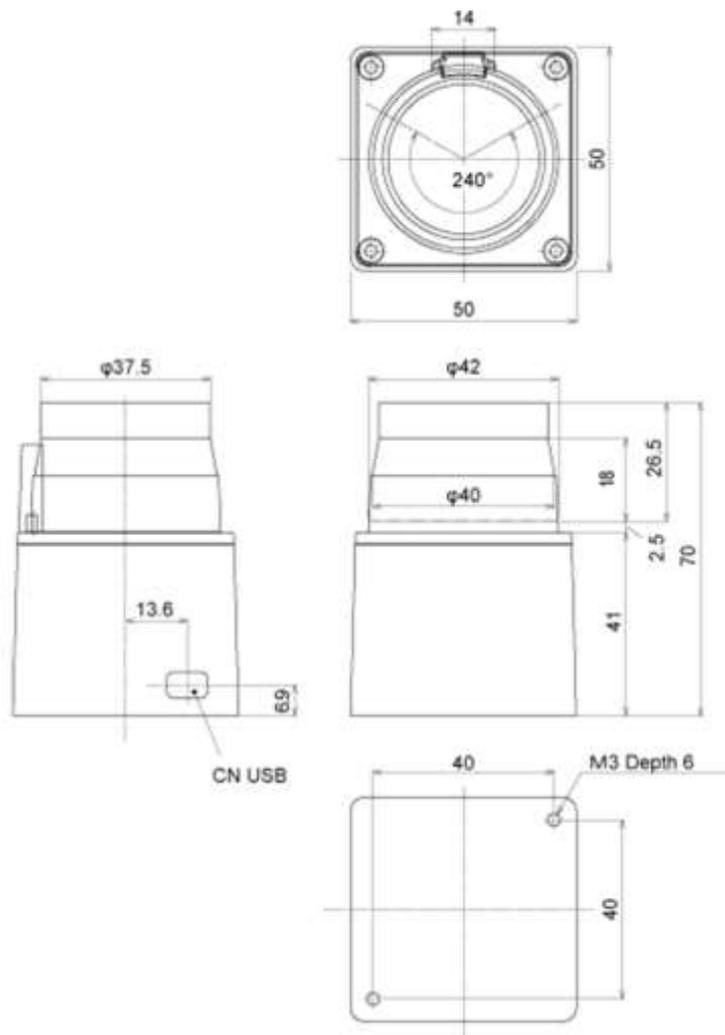
**Figura 30:** Sensor láser URG-04LX-UG01

Cada lectura del sensor contiene las lecturas de distancias medidas en cada punto, con un vector de 683 valores). La conexión con el ordenador se realiza mediante USB.



**Figura 31:** Características del sensor

Las dimensiones externas se muestran en la Figura 32



**Figura 32:** Dimensiones externas del sensor

A continuación se detallan las especificaciones del sensor:

- Fuente de luz: Diodo semiconductor láser ( $\lambda=785$  nm)
- Precisión: 60 a 1,000mm :  $\pm 30$ mm; 1,000 a 4,095mm:  $\pm 3\%$  de la medida
- Interfaz: USB 2.0
- Tiempo de escaneo: 100 ms/escaneo
- Ruido: 25 dB o menos
- Luminosidad: 6000-10000 Lux
- Temperatura / Humedad: -10 a +50°C / 85% o menos
- Resistencia vibración: 10 a 55 Hz, doble amplitud 1.5mm cada 2 horas en X, Y y Z
- Peso: 160 gr
- Voltaje: 5V@0.5A
- Resolución angular: 240°
- Distancia: 5,6 m
- Frecuencia de escaneo: 10 Hz
- Consumo: 2.5 W

Para la captura de las imágenes, y buscando realizar un dispositivo de detección de bajo coste, se va a utilizar una cámara web sencilla (Figura 33). Posteriormente se puede sustituir esta cámara por una de mejor calidad, que permitirá una captura más nítida de la escena que repercutirá positivamente en la técnica propuesta.



**Figura 33:** Cámara utilizada para la toma de imágenes

## 2. Calibración: Validación inicial del uso de láser en agua

Para medir el funcionamiento del láser en diferentes situaciones, como grado de turbidez del agua, o el tipo de material en el que el sensor estaría colocado dentro del agua (Figura 34) se realizaron una serie de ensayos.



**Figura 34:** Imágenes durante los ensayos de medición del láser

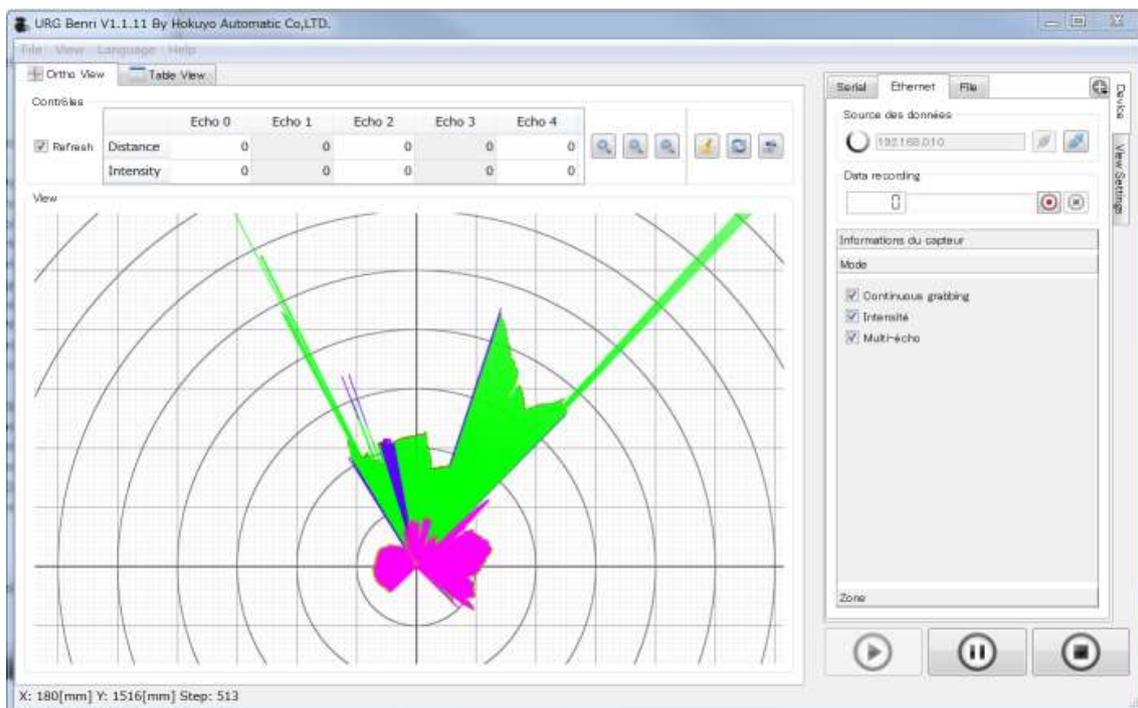
Para llevar a cabo las pruebas, se han dividido en dos fases. En primer lugar se ha usado agua limpia con el objetivo de evaluar el alcance que tiene el sensor láser en el

agua. A continuación, mediante el uso de limo de sílice, se añade turbidez al agua para reevaluar el alcance del sensor. Para provocar diferentes escenarios se establecen cuatro niveles de turbidez. El primero es con agua limpia. En los restantes se irá incrementando el aporte de limo de sílice progresivamente hasta conseguir que no haya nitidez para poder ver objetos directamente con una observación directa, este será el cuarto nivel.

Disponiendo de estas mediciones se puede conocer el comportamiento del láser en diferentes circunstancias y comparándolas con los valores reales, medidos manualmente, de la distancia entre el pez y el sensor laser, estableciendo una relación entre las dos mediciones.

Al no estar diseñado este sensor láser para ser sumergido en agua, se van a probar cuatro opciones para lograr la medición. La primera opción será con el sensor en el exterior del agua y pegado al cristal. Para las tres restantes se usará un recipiente diferente en donde introducir el sensor para sumergirlo. Estos recipientes serán un cilindro de metacrilato, otro de vidrio de cuarzo y un tercero de cristal.

El fabricante, *Hokuyo*, ofrece un software, *UrgBenri* (Figura 35), que permite visualizar los datos medidos por el sensor de una manera muy visual.

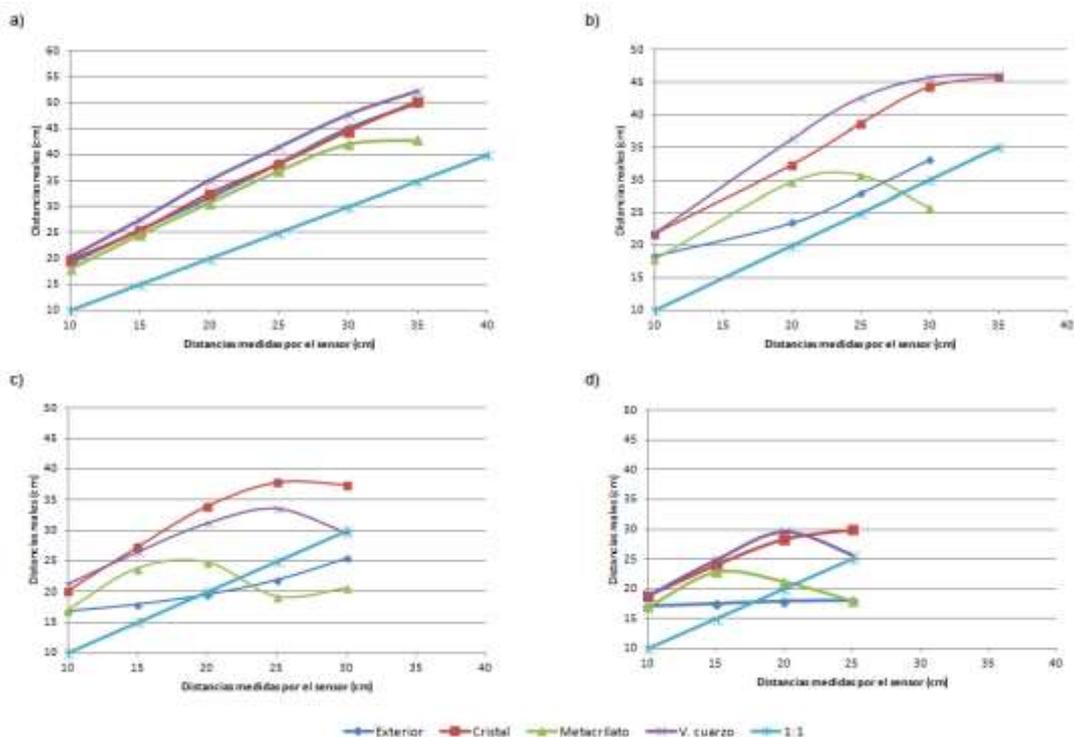


**Figura 35:** Captura de pantalla del software *UrgBenri*

En la Figura 36 se muestran las mediciones realizadas. En el eje vertical se mide la distancia real, en centímetros, a la que está el objeto. En el eje horizontal, y expresadas también en centímetros, se posicionan las medidas devueltas por el sensor láser

Según se observó en los experimentos, y se puede comprobar en las gráficas, se comprobó que, en el medio acuático, el sensor se comporta devolviendo una medida mayor que la distancia real a la que se encuentra el objeto. Además, llega una distancia en la que se empieza a perder precisión. Esto debe ser tenido en cuenta para llevar a cabo el cálculo de las distancias a las que se encuentre un objeto detectado por el sensor para que, de esta forma, se pueda estimar la distancia de la forma más acorde con la distancia real.

Otro aspecto que se puede ver en las gráficas es el cómo se comporta el sensor láser según el recipiente usado y según el nivel de turbidez. Así, los materiales que permiten medir distancias más lejanas correctas son el cristal y el vidrio de cuarzo, mientras que el metacrilato muestra un comportamiento algo peor.



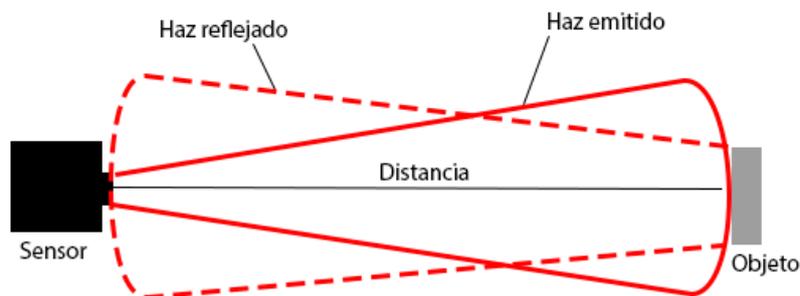
**Figura 36:** Mediciones realizadas para analizar el comportamiento del láser en agua utilizando los niveles de turbidez (a) agua limpia, (b) nivel bajo-medio, (c) nivel medio-alto y (d) nivel alto

Con la validación y el conocimiento del comportamiento del láser en el agua, en el siguiente punto se procede a detallar las diferentes fases en las que se va a desarrollar la técnica propuesta.

### 3. Transformación de la información capturada

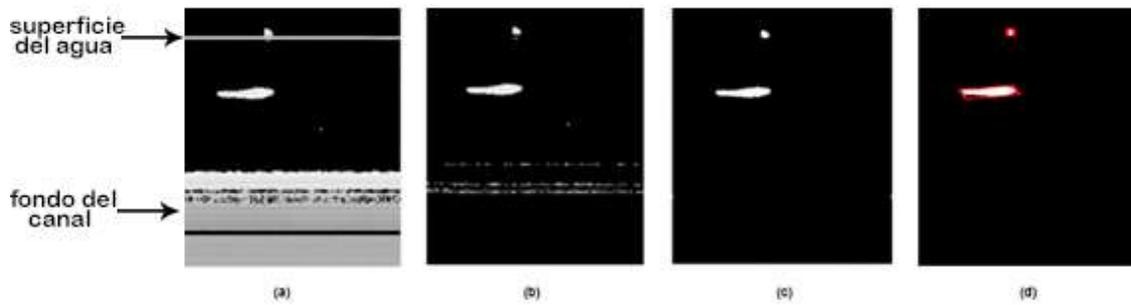
El principio básico del sensor láser es el envío de un pulso de haz láser y medir el tiempo que tarda ese pulso en ser reflejado en el objeto y devuelto al sensor. Ese valor se utiliza para saber la distancia a la que se encuentra el objeto haciendo uso de la ecuación (24). Este funcionamiento se muestra en la Figura 37.

$$Distancia = \frac{Velocidad \times Tiempo}{2} \quad (24)$$



**Figura 37:** Cálculo de distancias mediante un sensor láser.

En el punto donde se desea detectar el paso del pez se sitúa el sensor, que irá devolviendo a su salida puntos de medida que serán codificados y, de esta forma, se podrán representar en una escala de grises (Figura 38a). Según el grado de gris en la escala se indicará cuánto de cerca o de lejos está el objeto con respecto al láser: cuanto más lejano, el punto se acercará más al negro. Por el contrario, cuanto más cercano al sensor, más blanco aparecerá en la imagen. Hay que reseñar que en el sensor utilizado, las medidas se espacian cada  $0.35^\circ$  sobre el rango de  $240^\circ$ , obteniendo un vector de 683 valores.



**Figura 38:** Detalle de las diferentes etapas en el proceso de visión artificial. (a) imagen generada (b) imagen después del pre-procesamiento (c) imagen al término del proceso de segmentación (d) imagen con los objetos candidatos seleccionados

Con la obtención de la imagen, se puede iniciar el proceso de visión artificial, que estará dividido por etapas: pre-procesamiento, segmentación y reconocimiento e interpretación.

#### 4. Pre-procesado

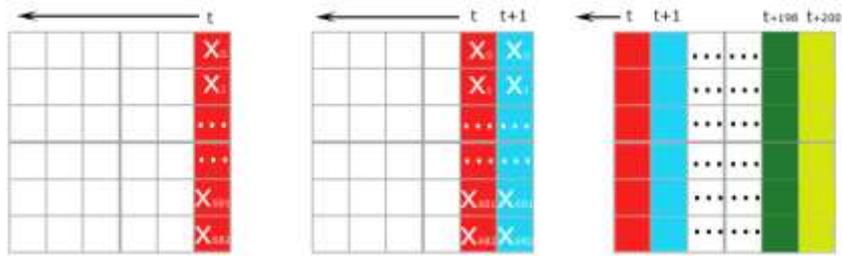
Para mejorar la calidad de la imagen se inicia el proceso con un pre-procesamiento de la imagen. Este proceso facilitará etapas posteriores, ya que los tratamientos y análisis sobre la imagen resultante serán menos costosos que con la imagen original.

Lo primer que se realiza es una eliminación del fondo. Para ello se va a utilizar una captura del escaneo del láser en un momento en el que ningún objeto en movimiento sea interceptado por el láser y sobre cada nuevo vector de datos mandado por el sensor se aplica la resta de ese vector libre de objetos. Los píxeles que no sufran variaciones, que va a tener valor cero tras la sustracción del fondo, se representan en negro (Figura 38b).

#### 5. Recepción de la información en tiempo real

Cuando se recibe un vector de entrada, automáticamente se guarda en un buffer asignando los correspondientes valores de gris, según se comentó anteriormente. El sensor envía datos de lectura cada 100 milisegundos que son almacenados en el buffer

por el final del mismo, moviendo el resto de los vectores una posición hacia delante, tal y como se ilustra en la Figura 39. Con este procedimiento se consigue una imagen en escala de grises del área que cubre el láser y que se actualiza de manera automática y en tiempo real, por lo que simula el comportamiento de un vídeo.



**Figura 39:** Procesamiento del buffer de lecturas enviadas por el sensor cuando recibe una nueva entrada.

## 6. Detección de objetos

Para realizar la detección de objetos se realizará una segmentación sobre la imagen. Esta etapa es fundamental en los sistemas automatizados de visión artificial tanto por los resultados, que son muy importantes para continuar el proceso de visión artificial, como por las dificultades que conlleva conseguir esos resultados. Mediante la segmentación, se consigue la división de la imagen en las partes o los objetos que la forman. Así, se puede diferenciar entre objetos no relevantes (que serán considerados como parte del fondo) y objetos de interés.

Dentro de las técnicas de segmentación se puede realizar una clasificación en 3 grupos: en primer lugar están las técnicas basadas en la detección de la frontera, en segundo lugar las técnicas de umbralización y por último las que son basadas en el agrupamiento de píxeles. En el caso de este trabajo, sobre la imagen resultante se aplica la técnica de umbralización binaria que, gracias a su coste computacional bajo y a su rapidez, permitirá que esta etapa se complete en tiempo real.

En la imagen de entrada lo que interesa es separar cualquier objeto detectado por el sensor láser, sin importar su posición. Para ello, se elige un elemento de la imagen que

no sea negro, y se establece un umbral suficiente que permita filtrar los elementos más alejados, y se establece su valor en blanco, obteniendo así una imagen binaria.

Buscando una optimización de los resultados, se aplican operaciones morfológicas [82] sobre la imagen. Las operaciones morfológicas se engloban dentro del marco teórico para el procesamiento digital no lineal de imágenes, conocido como la morfología matemática. A través de su utilización se cuantifican muchos aspectos de las estructuras geométricas que se encuentran en una imagen, de forma que coincidan con la percepción humana y la intuición [83].

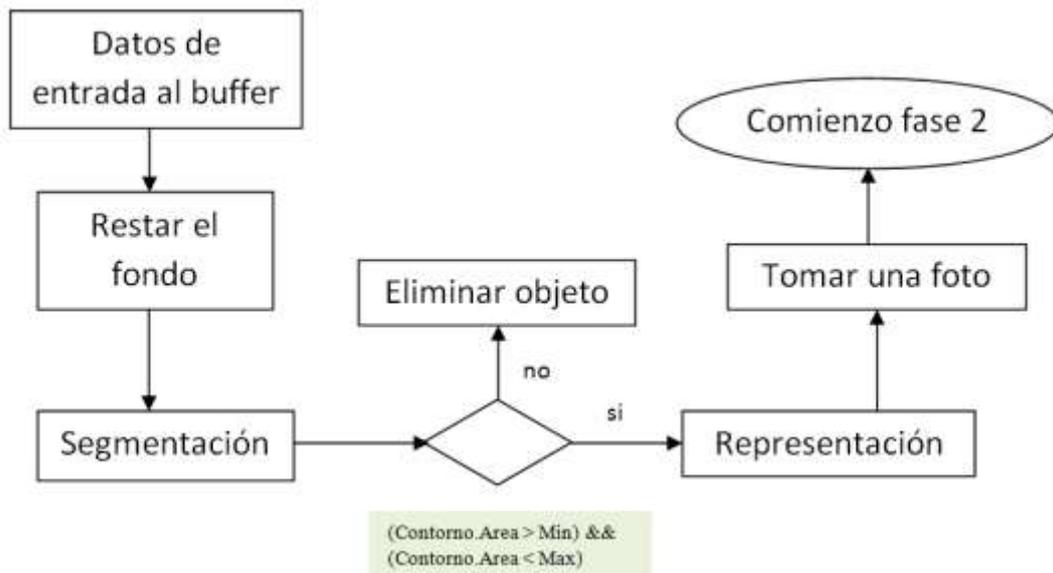
Dentro de las operaciones morfológicas que existen, se van a utilizar las aperturas y cierres [82, 83]. Ambas operaciones buscan suavizar el contorno de aquellos elementos sobre las que se aplican. El cierre realiza un relleno de los huecos que aparecen en el objeto, para reducirlos o eliminarlos. La operación de apertura elimina aquellos objetos más pequeños y también permite separar los que se encuentran cercanos. Para conseguir los efectos deseados, sobre la imagen se realiza un proceso consistente primeramente en aplicar una apertura y, a continuación, sobre el resultado se aplicará 3 veces una operación de cierre. Estas operaciones comentadas anteriormente se realizan con la cruz como elemento estructural. Con esta combinación de operaciones morfológicas, se realiza un filtrado y una optimización de la imagen, pensando en las fases posteriores (Figura 38c).

Terminada la fase de segmentación se inicia la nueva fase consistente en la representación y descripción de los objetos que se han segmentado de la imagen en la fase anterior. Con esta fase se busca, por un lado, elegir la caracterización de los segmentos que permitan una mejor manipulación de los mismos y, por otro, encontrar los rasgos descriptores de estos segmentos y que estos rasgos, ante transformaciones geométricas, se mantengan lo más estable posible.

Para conocer los rasgos geométricos básicos, como puede ser el área del objeto o su alto y ancho se engloba el objeto dentro del mínimo rectángulo posible. Se consigue así una caracterización de los diferentes segmentos de la imagen, gracias a una descripción externa que, en este caso, son los píxeles que componen el contorno de ese segmento. Según las características extraídas, o bien se descarta el objeto detectado, o bien se continua el proceso de visión artificial (Figura 38d). Para descartar objetos, en los

experimentos realizados, se establece que el área del contorno del segmento estudiado se encuentre dentro del intervalo [40, 1000].

En la Figura 40 se ilustra esta primera fase de la práctica propuesta, esquematizando las diferentes partes del proceso de visión artificial realizado.



**Figura 40:** Esquematización de la primera fase de la técnica propuesta.

La segunda fase de la técnica propuesta es la fase de verificación o rechazo del objeto detectado. Sobre el objeto detectado y marcado como candidato a pez, que ha sido detectado en la fase anterior, se debe verificar esa posible condición de pez. Para ello, en el momento en el que el láser detecta el objeto, la cámara, situada para cubrir la zona de escaneo del láser, toma una captura de la zona. En los ensayos se ha situado una cámara web, que tiene una resolución VGA, que devuelve una imagen RGB.

Con la imagen que se obtiene se procede a iniciar un nuevo procedimiento con técnicas de visión artificial formado por las mismas etapas que en la fase anterior, esto es, etapa de pre-procesamiento, de segmentación y de representación e interpretación.

Al igual que la primera fase de la técnica anterior, el pre-procesamiento va a consistir en una sustracción del fondo. Para ello, lo primero que se hace es tener almacenada una muestra de la zona de captura de la cámara sin ningún objeto en

movimiento interceptado por el láser. Esta muestra debe ser actualizada cada poco tiempo con el objetivo de reducir la influencia que pueda tener en la técnica situaciones como el cambio de luminosidad de la escena. Con esa muestra, se realiza la operación de substracción sobre la imagen, mostrando así la parte de la imagen con información relevante.

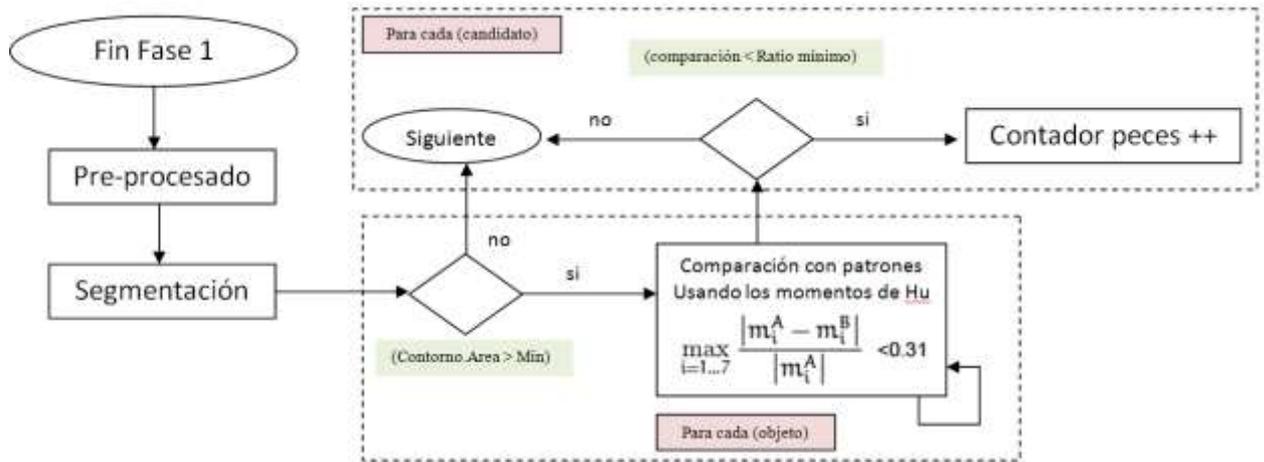
Para conseguir una buena segmentación, se convierte la imagen obtenida a escala de grises y seguidamente se realiza una umbralización binaria. Para seleccionar un umbral óptimo en la separación del fondo y el contenido, se utiliza un método automático de cálculo conocido como método de Otsu [84].

Seguidamente se tiene que determinar si se pueden considerar como peces los objetos que hayan sido segmentados, por ello se inicia la etapa de representación e interpretación. El primer paso es eliminar los objetos que no cumplen ciertos criterios establecidos, como por ejemplo que el área del objeto sea menor de un determinado valor. En el caso de los experimentos realizados se establece el criterio de que un objeto será eliminado si su área es menor de 5000.

Para los objetos que queden se van a utilizar momentos espaciales [85], que son medidas estadísticas que según los puntos interiores de las regiones (en este caso los objetos), permiten describirlas. Concretamente se utilizan los siete momentos de Hu [86] que son momentos centrales normalizados de orden 2 y 3, y son invariantes a escalados, rotaciones y traslaciones.

Con el objetivo de poder comparar los objetos segmentados, en este caso se tienen establecidos diferentes patrones con formas de peces. Si la comparación entre el patrón y el objeto segmentado cae dentro de un intervalo determinado (en el caso de los ensayos realizados se establece a 0,31), se acepta el objeto como pez, se incrementa el contador del número de peces y se marca su contorno. En caso contrario, el objeto segmentado es rechazado como candidato a ser un pez.

El proceso completo de esta segunda fase de la técnica propuesta se resume en la Figura 41.



**Figura 41:** Esquematización del algoritmo utilizado.

## 7. Seguimiento de objetos

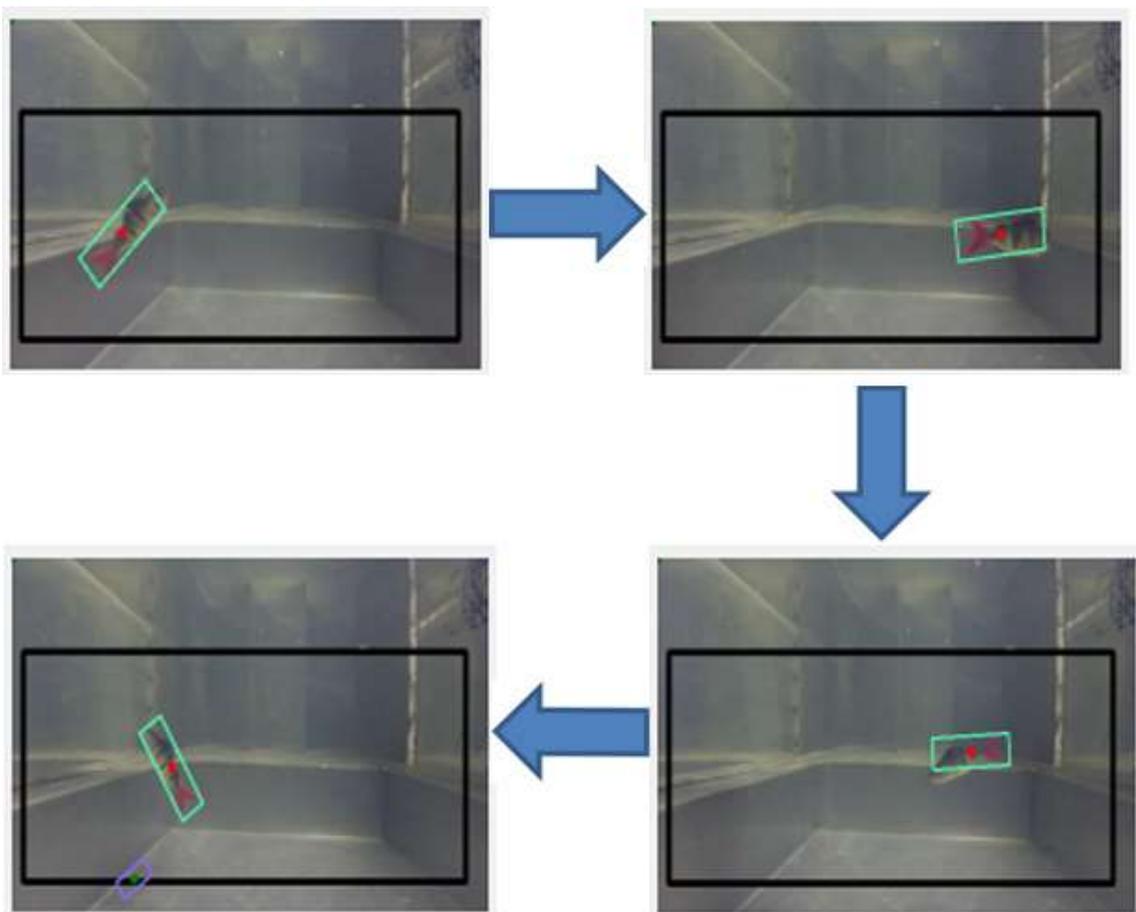
Se desarrolla un sistema de seguimiento de las detecciones con el fin de reducir el conteo de posibles duplicados, ya que si no se realiza este seguimiento el mismo pez podría ser contado múltiples veces en imágenes consecutivas. Este seguimiento consigue su objetivo mientras el objeto que monitoriza no salga de la ventana de visualización. Una vez esto se produce, es imposible determinar si es el mismo pez que estaba anteriormente o no, con lo que se contará como uno nuevo.

El funcionamiento consiste en almacenar los rasgos característicos de cada contorno que se detecta y que han sido calculados previamente, almacenando las posiciones y área del rectángulo mínimo que lo contiene, el punto central, y un color asignado para colorear su contorno. Se dispone de un contenedor de detecciones que se irá refrescando según aparezcan o desaparezcan detecciones.

Dentro de ese contenedor se comparan las nuevas detecciones con las detecciones antiguas estimando que la posición del punto central de la detección debe estar dentro de una ventana de búsqueda pequeña, que la forma debe tener un área similar y las dimensiones del rectángulo también deben ser semejantes. En caso de que no se cumplan las condiciones de comparación, se añade la nueva detección al contenedor de detecciones y en caso de que ya exista, no se añade, pero si se marca como que ha vuelto a aparecer.

Esto se hace ya que si una detección no aparece dentro de un tiempo determinado, se considera que esa detección ya no se encuentra en la imagen y se elimina del contenedor.

En la Figura 42 se muestra una secuencia de seguimiento de un pez en la imagen. El pez aparece mostrado dentro del rectángulo verde y se realiza un seguimiento de él hasta que desaparezca de la ventana de visualización (marcada en negro).



**Figura 42:** Proceso de seguimiento de un pez

## 8. Obtención de mediciones del objeto

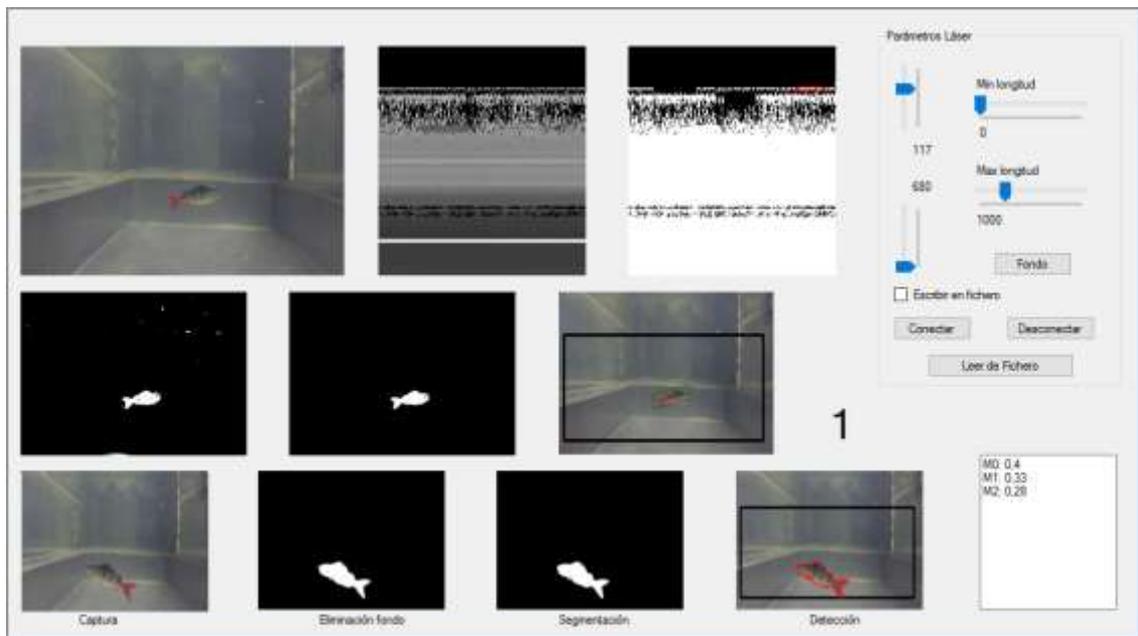
En el caso práctico anterior se necesitaba la visión estéreo para ser capaz de generar un mapa de profundidad a través del cual poder estimar el tamaño de los peces. En este caso se dispone del sensor láser que permite conocer directamente la distancia a

la que se encuentra el pez. Por ello, en este caso la medición se realiza de forma directa, teniendo la imagen en donde aparece el pez, sabiendo la distancia del sensor al pez (lectura del láser), la distancia del sensor a la cámara y habiendo realizado diferentes medidas de patrones conocidos previamente a modo de calibración.

## 9. Visualización en tiempo real

Para la visualización se ha desarrollado un software que ilustre el proceso completo de la técnica y que permite modificar los parámetros tanto de configuración del láser como de la detección.

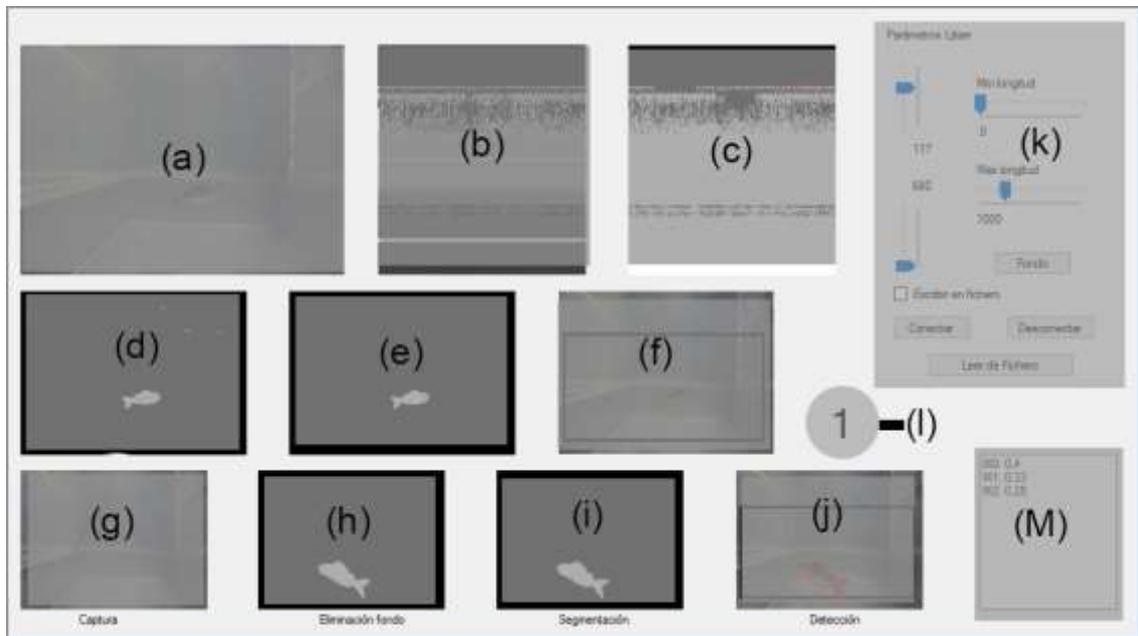
La pantalla principal de la aplicación muestra el proceso completo de la detección, juntando la fase de detección por láser y la fase de identificación mediante toma de imágenes (Figura 43).



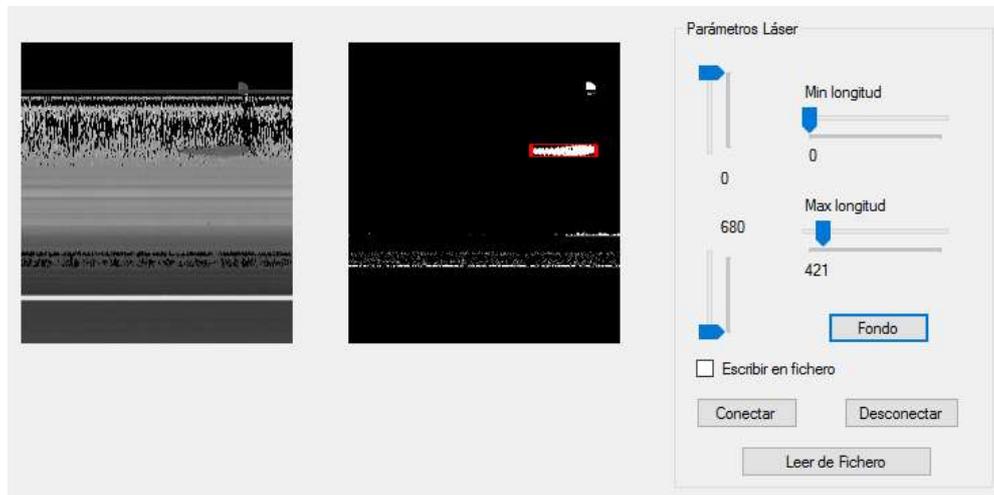
**Figura 43:** Pantalla principal de la aplicación

Lo relativo al sensor láser se encuentra en la zona superior derecha (Figura 45), mientras que el resto de la ventana hace referencia al proceso de identificación mediante las imágenes recibidas. Esta zona está formada en primera instancia por la imagen generada con las lecturas del láser y en escala de grises (Figura 44b). A su derecha está

la visualización de la imagen pero esta vez en formato binario (Figura 44c), que será la imagen con la que se trabaje en el proceso de visión artificial.



**Figura 44:** Mapa de la pantalla principal



**Figura 45:** Zona de parámetros láser

A su derecha está la zona de gestión de parámetros del láser (Figura 44k). Mediante el desplazamiento de las barras de valores mostradas en la figura se puede modificar la zona de detección del láser, tanto en altura como en profundidad (longitud). Debajo de las barras de modificación de la longitud se encuentra el botón “Fondo” que permite tomar una muestra de la lectura del sensor en ese momento y eliminar de la

imagen todo lo que permanece inmóvil. Debajo de las barras de modificación de la altura del láser aparece un *checkbox* que debe estar marcado si se desea que la lectura que se recibe del sensor láser se almacene en un fichero para su posterior revisión.

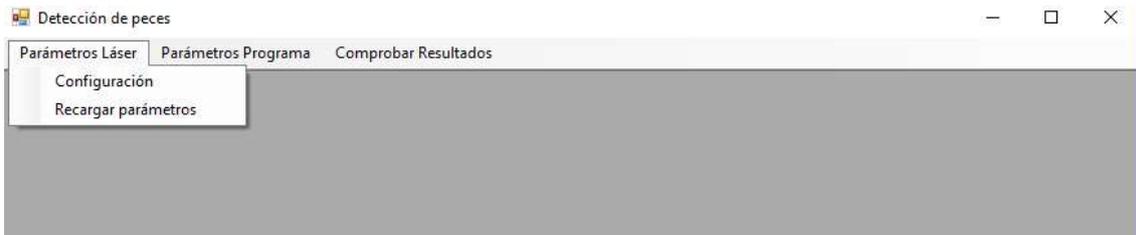
En la parte inferior de la zona de gestión de parámetros del láser se encuentran los botones que permiten la conexión y desconexión de la aplicación al sensor láser, así como el botón para cargar un archivo de texto con los valores de una lectura previa.

El resto de la pantalla principal está relacionado con la identificación del objeto que detecta el sensor láser. La primera imagen que aparece en la pantalla (Figura 44a), es la visualización de la imagen de entrada de la aplicación, ya sea de una cámara o de un vídeo, según lo que el usuario haya elegido.

En la siguiente fila hay tres imágenes que se visualizan de forma continua, al igual que la anterior. La primera (Figura 44d) es el resultado del proceso de eliminación de fondo, a su derecha se muestra el resultado después de la segmentación (Figura 44e) y la última (Figura 44f) es la que muestra el proceso de seguimiento. A la derecha de esta imagen, se encuentra el contador de peces detectados (Figura 44l).

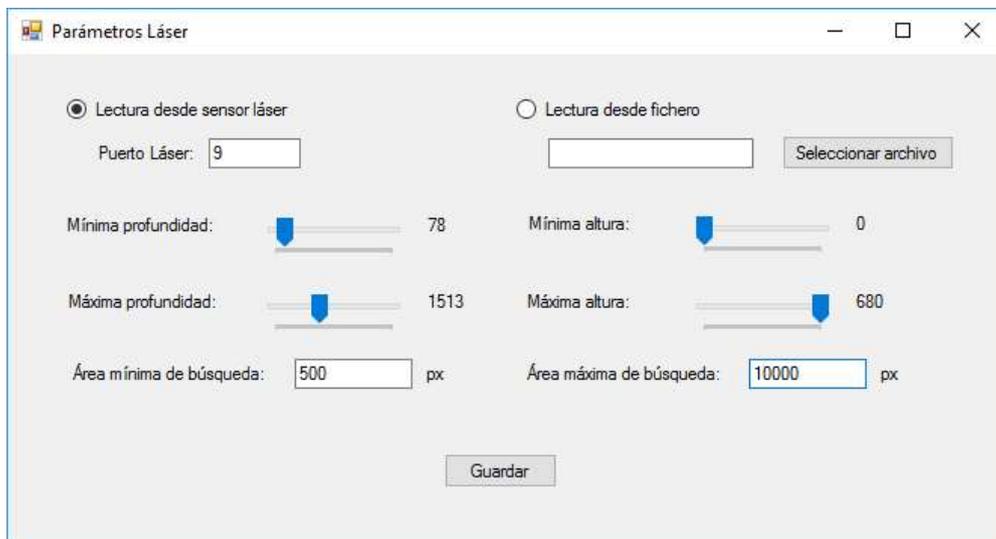
En la última fila se muestran las capturas del resumen del último proceso de identificación. La primera captura (Figura 44g) es la imagen que hay en la entrada cuando el láser hace la detección. La segunda imagen (Figura 44h) es la imagen tras la eliminación de fondo y a la entrada del proceso de segmentación. La tercera es la salida del proceso de segmentación (Figura 44i) y la última es la visualización del resultado sobre la imagen real (Figura 44j). A la derecha de esta última imagen aparece un cuadro con los resultados de la comparación de la detección con los diferentes patrones de los que se disponen (Figura 44m).

Los parámetros de configuración del láser se pueden modificar desde el menú *Parámetros Láser* de la aplicación (Figura 46). Este menú permite acceder a la pantalla de configuración o recargar la configuración que esté establecida.



**Figura 46:** Menú de la sección “Parámetros Láser”

Las modificaciones realizadas en esta pantalla serán guardadas para cargar por defecto en posteriores aperturas de la aplicación y serán cargados también en la instancia actual. Las modificaciones realizadas en la zona de parámetros láser de la pantalla principal (Figura 44k) podrán ser guardados antes de cerrar la aplicación.



**Figura 47:** Pantalla de modificación de parámetros del láser

Para recibir la entrada del sensor láser se debe seleccionar, en primer lugar, en la pantalla de configuración de los parámetros del láser (Figura 47), si la entrada será directamente recibida del sensor o bien será a través de un fichero. En el momento en que se cambie esta selección, la aplicación deshabilitará los botones correspondientes. En caso de ser a través del sensor, el usuario debe introducir el puerto COM al que estará conectado el sensor. En caso de ser a través de un fichero de texto, el usuario elegirá la localización del archivo.

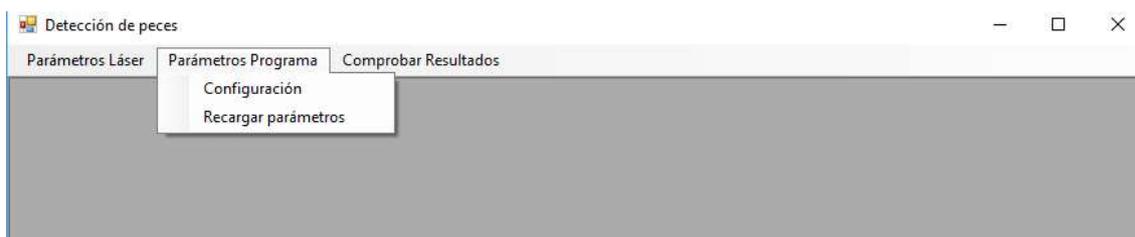
El usuario dispone de cuatro controles para modificar la altura y profundidad, dos para cada dimensión. Para realizar esta modificación el usuario solo deberá desplazar el

indicador de la barra. Estos controles se pueden encontrar tanto en la pantalla principal (Figura 45) como en la pantalla de parámetros (Figura 47).

Los últimos parámetros configurables del láser son el área mínima y máxima de búsqueda. Esto es el tamaño en píxeles del área de los contornos que se desean estudiar.

Una vez elegida la configuración deseada, ya se puede empezar a recibir datos relativos al láser. Para ello el usuario, desde la zona del láser (Figura 45) de la pantalla principal (Figura 44k) pulsará el botón "*Conectar*" y para detener la recepción de lecturas, pulsará el botón "*Desconectar*", en el caso de lectura desde el sensor.

Los parámetros de esta sección se pueden modificar desde el menú *Parámetros Programa* de la aplicación (Figura 46). Este menú permite acceder a la pantalla de configuración o recargar la configuración que esté establecida.



**Figura 48:** Menú de parámetros del programa

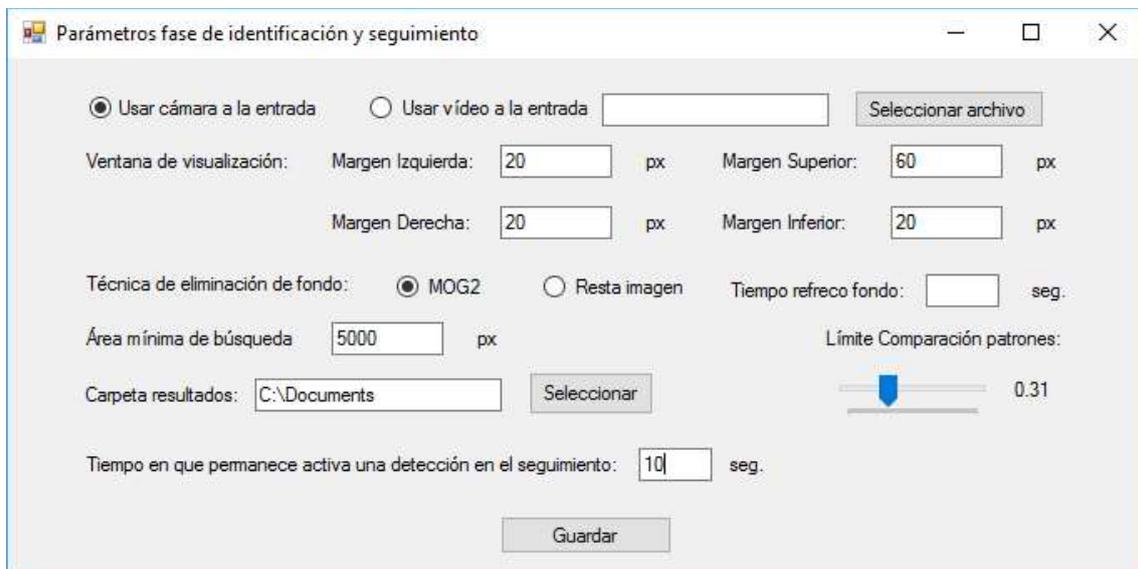
En la Figura 49 se pueden ver los diferentes parámetros que se pueden configurar. La primera fila indica el tipo de entrada que se va a tener, si desde una cámara o desde un fichero. En la segunda fila se permite modificar los márgenes que se reducirán del cuadro de entrada, para realizar las identificaciones solo de los objetos que estén dentro de la zona restante.

La tercera fila permite seleccionar el tipo de eliminación de fondo. En caso de elegir "*Resta imagen*" se debe seleccionar el tiempo de refresco que se desea para esa imagen de fondo que se utiliza para la substracción.

La cuarta fila es el valor en píxeles del área mínima que deben de tener los contornos tras la segmentación para ser tenidos en consideración. Debajo de esto aparece

la opción de elegir la carpeta donde se quiere que la aplicación almacene los resultados del proceso de identificación. Y a su derecha se permite establecer el límite que debe cumplir la forma identificada para ser vinculada a un patrón.

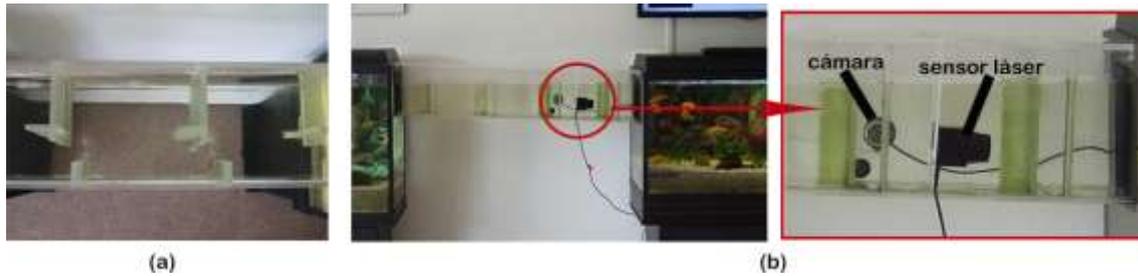
Por último se permite seleccionar el tiempo que se desea que un objeto detectado sea tenido en cuenta en el seguimiento, una vez salga fuera de la ventana de visualización o deja de ser detectado. Si ese tiempo se cumple, cuando el objeto sea nuevamente detectado, el sistema lo computará como un objeto nuevo.



**Figura 49:** Pantalla de modificación de parámetros de la identificación y seguimiento

## 6.3. Resultados experimentales

Se han realizado ensayos para poder evaluar el software de visión artificial desarrollado. Para los ensayos se ha construido una escala con metacrilato (Figura 50a), que une dos peceras (Figura 50b) que albergan una docena de peces con varios colores y diferentes tamaños. Esta escala de metacrilato es un modelo a escala 1:8,3.

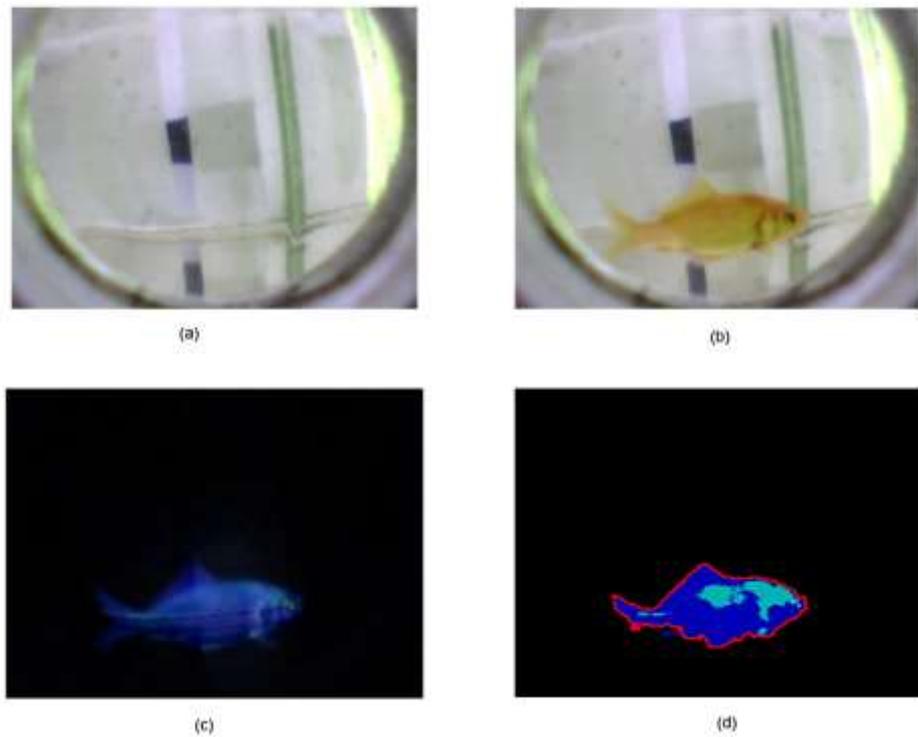


**Figura 50:** (a) Escala de metacrilato y (b) montaje de la cámara y el láser.

Se llevan a cabo diferentes experimentos en donde el sistema queda funcionando durante suficiente tiempo para que suceda una cantidad considerable de paso de peces por la escala. En cada detección que hace el láser se guardan las imágenes para poder, posteriormente, realizar un estudio.

También se realiza un ensayo con el objetivo de medir falsos positivos. Para este ensayo se buscan elementos de tamaño similar a los peces, como trozos de madera, de acero o de metacrilato, y con ayuda de un hilo se pasan por la zona de escaneo del láser y se guardan las imágenes para comprobar el funcionamiento del sistema.

El resultado del proceso seguido con la utilización de las técnicas de visión artificial para detectar el pez en la imagen tomada por la cámara se muestra en la Figura 51. Partiendo de la imagen de fondo (Figura 51a) y de la imagen capturada (Figura 51b) y después llevar a cabo la eliminación de fondo y la segmentación (Figura 51c) se puede llegar a la conclusión de si el objeto es un pez o no (Figura 51d).



**Figura 51:** Resultado del proceso de visión artificial sobre la imagen capturada por la cámara.

Para analizar los resultados, las imágenes fueron revisadas por expertos. En total se revisaron cerca de 200 imágenes correspondientes a la captura de la cámara cuando un objeto pasa por el punto de la escala en donde está situado el láser (Tabla 2). Con el objetivo de comprobar la fiabilidad del software desarrollado se desecharon las imágenes en las que el objeto no aparecía de forma completo o bien aquellas en las que el objeto queda fuera de la toma. Esto será realizado de forma automática en versiones posteriores.

Para medir el sistema, las métricas que se utilizan son las mostradas en las Ecuaciones (25), (26), (27) y (28):

$$\text{Precisión} = \frac{\text{Positivos correctos}}{\text{Positivos correctos} + \text{Falsos positivos}} \quad (25)$$

$$\text{Cobertura} = \frac{\text{Positivos correctos}}{\text{Positivos correctos} + \text{Falsos negativos}} \quad (26)$$

$$\text{Ratio de Falsos positivos} = 1 - \text{Precisión} \quad (27)$$

$$\text{Ratio de Falsos negativos} = 1 - \text{Cobertura} \quad (28)$$

<i>Métrica</i>	<i>Datos</i>	<i>%</i>
<i>Precisión</i>	0.97	97.18
<i>Cobertura</i>	0.73	72.63
<i>Tasa de Falsos Positivos</i>	0.03	2.18
<i>Tasa de Falsos negativos</i>	0.27	27.36

**Tabla 2 :** Resultados del ensayo

## 6.4. Conclusiones

Tras analizar los resultados que se obtienen en el análisis de los materiales en los que se introduce el sensor láser para sumergirlo, se puede llegar a la conclusión que para llevar a cabo operaciones que requieran sumergir el sensor, debería de utilizarse el cristal o el vidrio de cuarzo, ya que con estos materiales el haz del sensor alcanza las mayores distancias. También se deber tener en cuenta el efecto que se produce cuando se sumerge el sensor, y es que debido a la reflexión del haz del láser en el agua, se amplifican las distancias.

En la Tabla 2 se aprecia una alta precisión en el sistema. Esto indica que es capaz de detectar correctamente objetos en el 97% de los casos y una cobertura del 73% que detecta los peces. Como trabajo futuro se plantea la automatización de la eliminación de imágenes en donde el pez no aparezca completamente, evitando así la revisión del experto.

Mediante el uso de técnicas de visión artificial combinadas con la tecnología láser se ha desarrollado un sistema que permite detectar y contar peces. Este sistema se lleva a

cabo en dos fases, en la primera el sensor láser detecta el paso de un objeto, y en la segunda fase se trata de confirmar si ese objeto detectado es un pez o no.

Este sistema se lleva a cabo en tiempo real, es un método no invasivo para los especímenes y además no tiene el inconveniente de la limitación visual que pueden suponer algunas situaciones como por ejemplo la turbidez de agua.

Como principales ventajas de esta técnica se señalan el reducido tamaño del montaje necesario para realizarla, necesita solo un sensor láser, una cámara y un ordenador en donde ejecutar el software de procesamiento de visión artificial. Todo esto redundando en que es una técnica con un coste bajo que permite realizar el muestreo de la zona estudiada en tiempo real.

A pesar de que la técnica ha sido testada en un laboratorio, simulando situaciones adversas, debería analizarse en un contexto real para poder afinar más la detección de peces en entornos reales. También se debe evaluar cómo se comporta el sistema ante el paso simultáneo de varios peces por la escala de hendidura vertical.

# **7. Caso práctico 3: Detección de peces mediante luz estructurada**

## **7.1. Introducción**

Mediante la metodología propuesta se pretende desarrollar un sistema que permita la detección de peces bajo el agua utilizando escáner 3D basados en luz estructurada. Este tipo de dispositivos se basan en crear un patrón de puntos de luz infrarroja en un plano 2D. Mediante una fuente de luz infrarroja emiten el patrón de puntos preconfigurado sobre una superficie y capturan la refracción de los objetos situados en su campo de acción. Analizando las diferencias producidas en el patrón, obtienen una matriz de distancias a los objetos. Estos sistemas no están pensados para trabajar en entornos acuáticos y estos patrones de luz estructurada pueden tener un comportamiento completamente diferente. El objetivo de esta investigación es poner a prueba estos dispositivos para la detección de objetos bajo el agua, y desarrollar los algoritmos que permitan realizar las correcciones necesarias para que estos sistemas sean viables en estos campos de aplicación.

## **7.2. Metodología**

### **1. Dispositivo hardware de captura**

Para escanear objetos, existen múltiples dispositivos que se basan en el método de la luz estructurada. Para este caso práctico se ha utilizado un sensor desarrollado por Microsoft denominado KINECT. Dado que estos dispositivos se han desarrollado para entornos de videojuegos, se ha producido un importante abaratamiento en los costes, por lo que son muy accesibles y multitud de desarrollos software existentes actualmente. Este dispositivo ha aparecido en el mercado en 2010 (Figura 52) y posteriormente se ha

mejorado en una segunda versión en 2013 (Figura 53). En este caso práctico se han utilizado ambas versiones.



**Figura 52:** Kinect versión 1



**Figura 53:** Kinect versión 2

Estos sensores se basan en emitir un tipo de radiación (luz infrarroja en este caso) y generan una nube de puntos a partir del reflejo de dicha radiación en los objetos. Esta técnica es la más utilizada en los escáneres 3D de objetos, y necesitan un hardware especializado que inicialmente tenían un coste elevado. Pero dado que la industria de los

videojuegos los ha empezado a utilizar, se pueden adquirir por un coste bastante reducido. Como es el caso del Microsoft Kinect.

Actualmente en el mercado hay 2 tipos de dispositivos para escáner objetos 3D:

- Basados en técnica de vuelo (Time of Flight): se envía una señal óptica (por ejemplo láser o luz infrarroja) hacia la escena sobre la que pretende extraer la información 3D y la luz reflejada es detectada por el sensor, calculando el tiempo que tardó en ser de nuevo recibida (tiempo de vuelo). Como se conoce la velocidad de la luz, se puede calcular la distancia a la que se encuentran los puntos de la escena. Estos dispositivos (Figura 54) proporcionan medidas de alta calidad, estables en repetitividad y precisión. Debido a la necesidad de disponer de un hardware de altas prestaciones en tiempo de computación, tienen un coste elevado.



**Figura 54:** Dispositivo Time of Flight

- Luz estructurada. En este caso, los dispositivos emiten un patrón de radiación hacia los objetos y una vez reflejada obtiene la distancia de los objetos basándose en la deformación de dicho patrón emitido. En este caso, el sensor dispone de una calibración previa de dicho patrón. La Kinect se basa en este tipo de funcionamiento y el modelo de patrón es una nube de puntos (Figura 55).



**Figura 55:** Nube de puntos que genera la Kinect

En la Kinect se produce una nube de puntos como patrón, y el dispositivo produce una imagen con los valores de distancias a los objetos en cada pixel. La Kinect, produce en tiempo real 3 tipos de imágenes, una imagen en color, con una cámara convencional basada en RGB integrada en el propio dispositivo, una imagen de infrarrojo, de resolución 640x480 a una velocidad de 30 imágenes por segundo, que es donde se basa la captura de distancias, y una tercera que es la imagen de profundidad. En esta última, el dispositivo, produce una imagen de 640x480 de resolución a una velocidad de 30 imágenes por segundo, en escalas de grises donde cada píxel representa una codificación de la distancia de un objeto en esa posición. Este tipo de imagen tiene unas características que permite diferenciar regiones en la imagen:

- Regiones de medición: detecta un objeto y en los píxeles produce valores entre 0 y 255, que se puede trasladar a una codificación en escala de grises.
- Regiones de no medición: bien porque los objetos puedan estar fuera del rango de medición o porque existen objetos que ocultan zonas, donde no se alcanza el patrón de infrarrojo. En ambos casos el valor de los píxeles es 0.

Existen multitud de algoritmos que a partir de las imágenes de infrarrojos producen una matriz de puntos de profundidad. Sin embargo, la gran ventaja de estos dispositivos es que estos algoritmos están integrados en el propio hardware y producen en tiempo real una imagen en escala de grises con la codificación de profundidad con una resolución de 16 bits (rango de 65536 valores para la profundidad). Utilizando el hardware de la Kinect las características que se han obtenido de rango y precisión son las siguientes:

Rango de detección de objetos entre 50 cm y 5 metros. Puede llegar a 40 cm pero se pierde campo de visión. La resolución es de 640 x 480 con un ángulo de apertura de 45 grados en vertical y 58 grados en horizontal. Un análisis de geometría nos indica que tenemos sobre 0.75 mm por pixel ( $x, y$ ) a una distancia de 50 cm y sobre 3 mm por pixel a una distancia de 2 metros. Estos nos daría una resolución de profundidad variable, sobre 1.5 mm a 50 cm, y sobre 5 cm a la distancia máxima de 5 metros. Todos estos análisis están realizados en las condiciones de funcionamiento indicadas por la Kinect. Sin embargo, el patrón de puntos de infrarrojo interno de la Kinect está calibrado para medir al aire, es decir fuera del agua y dado que se va a utilizar para detectar objetos en el agua este es un medio mucho más denso, este procedimiento debe ser recalibrado. Como no podemos tener acceso al firmware interno de la Kinect, el procedimiento a seguir es desarrollar un algoritmo que procese los datos obtenidos de la Kinect para adaptarlos a la medición de objetos bajo el agua. Para ello, de forma experimental se utilizarán objetos bajo el agua colocados a diferentes distancias y diferentes ángulos de inclinación en los ejes vertical y horizontal, para producir una matriz de calibración que permita corregir las diferencias de medición producidas por el cambio de medio.

Mediante la colaboración en la dirección de un Trabajo Fin de Grado de Ingeniería Informática [87], se ha desarrollado un entorno de prueba para la calibración y desarrollo de los algoritmos necesarios. Se ha creado un entorno acuático artificial que consta de una pecera llena de agua en la que se introducen diferentes objetos junto con peces artificiales. Para obtener un fondo heterogéneo se coloca una cartulina recortada de diferentes colores en la parte posterior de la pecera. Los peces artificiales introducidos en el entorno pueden ser colocados estáticamente o movidos con sedal para simular el movimiento real de un pez.

La pecera utilizada tiene unas dimensiones 40x24x28 cm. Se han utilizado una serie de peces artificiales blandos con movilidad, que son utilizados generalmente como señuelo de pesca, para simular peces reales. Algunos de los peces tienen colores y forma similar al pez original, pero otros tienen forma estándar y tan solo los colores del original. La razón principal de la utilización de peces artificiales es para facilitar la simulación de diferentes situaciones sin necesidad de interactuar con animales vivos y causarles ningún tipo de daño o estrés. Una vez desarrollado el sistema completo se probará en entornos reales con peces vivos. A continuación se detalla la técnica seguida en el trabajo [87].

## 2. Calibración

La calibración se realiza para conocer cómo responde el frame de profundidad ante el entorno que analiza. Su objetivo es formalizar los rangos de funcionamiento del sensor, poder interpretar sus valores, conocer el margen de error, como es la relación con la salida del frame de infrarrojos y cómo reflejan los objetos el patrón emitido, ya que si no tiene un funcionamiento acorde lo esperado pueden producirse que algunas zonas no sean alcanzables por el sensor, o que solo aparezcan algunas veces.

Se ha realizado una calibración para establecer el significado de los valores y el rango real del sensor, para esta tarea se ha colocado la Kinect frente a varios objetos diferentes y se ha comprobado la distancia real con las mediciones obtenidas. Es necesario identificar cómo cada píxel del frame de infrarrojos corresponde al mismo píxel del frame de profundidad, para ello se realiza una identificación sobre diferentes tipos de objetos. Una vez identificados se pueden realizar procesos de calibración con objetos bajo el agua.

En una calibración con objetos fuera del agua (proceso para el que está configurado la KINECT) se tiene en cuenta la distancia y el ángulo con respecto al objeto, pero en este caso esto no es suficiente porque hay más elementos que influyen en la medición, como pueden ser el agua y el cristal. Debido a esto, se han llevado a cabo varios procesos de calibración modificando distancias y ángulos del sensor, así como haciendo pasar peces en diferentes ángulos y a diferentes distancias del sensor.

El objetivo principal de este procedimiento es encontrar una metodología que permita realizar el cálculo de la distancia real de un objeto que se encuentre en el interior del agua, considerando la distancia existente entre el objeto y el cristal de la pecera. Para ello se coloca un objeto a una distancia fija y conocida, y la Kinect a una distancia fija de la pecera (84 cm en este caso). Se han tomado las mediciones y capturas modificando la posición de la pecera con el fin de alterar la distancia que hay entre el objeto y el cristal, y variando así la cantidad de agua existente entre ambos. Algunos de los resultados obtenidos son los mostrados en la Tabla 3.

<i>Dist(Sensor,Cristal)</i>	<i>Dist(Objeto,Cristal)</i>	<i>Medición</i>	<i>Diferencia</i>
805mm	35mm	826mm	14mm
700mm	140mm	803mm	37mm
620mm	220mm	781mm	59mm

**Tabla 3 :** Distancia objeto: 840mm

A partir de esta tabla se puede obtener una ecuación de calibración previa para trasladar a distancia real la medición producida por la Kinect. Esta ecuación (29), a pesar de ser solo una aproximación en la que solo se han utilizado para las pruebas las zonas centrales de las imágenes, puede ser útil si se quieren hacer correcciones.

$$\begin{aligned}
 \text{Dist}(\text{Sensor}, P1) & \qquad \qquad \qquad (29) \\
 & = \text{Dist}(\text{Sensor}, \text{Cristal}) \\
 & + (\text{Medicion}(P1) - \text{Dist}(\text{Sensor}, \text{Cristal})) * 1,35
 \end{aligned}$$

El valor obtenido para la corrección, según las mediciones realizadas, se asemeja bastante al índice de refracción del agua (1,3330), parece que la causa principal de que la Kinect subestime la distancia es debido a la refracción debida al agua.

También es muy importante obtener una posición adecuada de la Kinect en relación con el cristal de la pecera para evitar lo máximo posible el reflejo de la luz infrarroja emitida. Es importante asegurarse de que no se produce un reflejo directo del

haz de luz infrarroja en la lente de la cámara infrarroja, esto se puede comprobar visualizando el frame infrarrojo y se soluciona girando levemente el sensor. Después de varias pruebas se ha considerado como distancia óptima entre 60 y 70 cm y completamente perpendicular al cristal. Estos valores también permiten maximizar el rango de medición de la Kinect, donde la distancia mínima al objeto debe ser de 50 cm.

### 3. Transformación de la información capturada

La Kinect va a proporcionar tres tipos de frames. Un frame de color RGB, uno infrarrojo y otro de profundidad. Estos frames se transforman en matrices de OpenCV para poder trabajar con ellos.

El frame infrarrojo es una representación en 8 bits de la intensidad de la intensidad de la luz infrarroja que se refleja en ese punto. Esta representación es en forma de matriz y se obtiene una nueva lectura 30 veces por segundo. El sensor emite un patrón de luz infrarroja que sirve para medir la distancia a los puntos de la imagen y este patrón y su variación es claramente visible en el frame de infrarrojo.

Ese patrón de luz emitido es también utilizado para calcular la distancia de cada uno de los píxeles que se han obtenido en el frame infrarrojo, generando así el frame de profundidad, que es el más característico en estos sensores. Esa información, la distancia, se codifica en 16 bits y se almacena en una matriz que tendrá una frecuencia de actualización idéntica al frame de infrarrojos, es decir, de 30 por segundo.

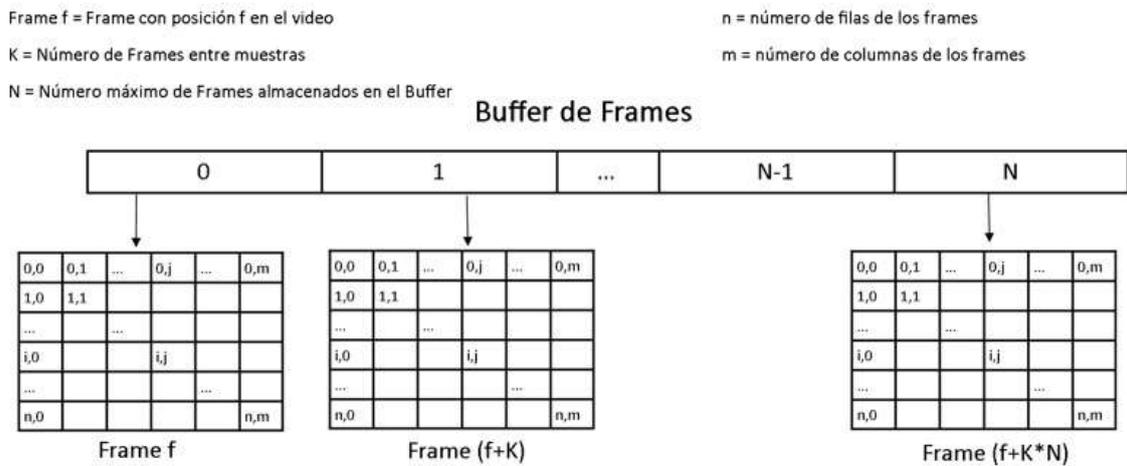
### 4. Pre-procesado

Para una correcta detección de un pez, es necesario eliminar aquellos objetos que están presentes de forma estática, tales como plantas, rocas u otros objetos que pertenecen al entorno y que se detectarían por la Kinect. De esta forma se puede mejorar el sistema de detección al reducir el tiempo de procesado dado que se eliminan aquellos objetos estáticos que están presentes de forma estática en las escenas. La sustracción del fondo es una técnica habitual en visión artificial. Mediante esta técnica se permite descartar gran

parte de la detección para centrar el procesado de detección de contornos y objetos en aquellos que aparecen nuevos en las escenas. En una detección bajo el agua, el fondo también puede ser cambiante, por lo que se aplicarán técnicas que permitan aprender la detección del fondo de forma dinámica. Existen diferentes algoritmos que permiten este procesamiento [88], y se adaptarán a la aplicación de sustracción del fondo, pero en este caso a las imágenes de infrarrojo y de profundidad que produce la Kinect.

*Técnicas de sustracción del fondo basado en aprendizaje con Buffer:*

Este tipo de aproximación está basada en la recolección de frames en instantes diferentes y el aprendizaje del fondo a partir de ellos. La obtención de frames se realiza en intervalos regulares, y estos son almacenados en un buffer, este buffer tiene un tamaño fijo y cuando es rebasado los frames más antiguos son sustituidos por los nuevos (Figura 56).



**Figura 56:** Funcionamiento del buffer de frames

El intervalo de obtención debe ser lo suficientemente alto como para que los peces que pertenecen a las escenas en frames sucesivos se encuentren en posiciones diferentes, pero hacerlo demasiado alto ralentizaría el refresco del buffer y con ello la flexibilidad del sistema, que aumentaría el tiempo de aprendizaje del fondo y el tiempo de adaptación a un cambio del fondo. Se ha optado por un tamaño de buffer de 30 frames y un intervalo de obtención de muestras de 30 frames, lo que equivale a un período de 1 segundo.

---

**Algorithm 1** ActualizarBuffer

---

```
1: intervaloCaptura ← 30
2: tamanoBuffer ← 30
3: cols ← 640
4: rows ← 480
5: numFrame ← 0
6: fondo ← ushort[rows, cols]
7: buffer ← ushort[][][tamanoBuffer]
8: procedure INTRODUCIRFRAME(frame)    ▷ Llamado cada vez que llega un nuevo
   frame
9:   numFrame ← numFrame + 1
10:  if numFrame mód intervaloCaptura = 0 then
11:    posicion ← (numFrame/intervaloCaptura) mód tamanoBuffer
12:    for row ← 0 to rows − 1 do
13:      for col ← 0 to cols − 1 do
14:        buffer[posicion][row][col] ← frame[row][col]
15:      end for
16:    end for
17:    ACTUALIZARFONDO()
18:  end if
19:  ...    ▷ El procedimiento realiza más operaciones con el frame
20: end procedure
21: procedure UPDATEBACKGROUND( )    ▷ Calcula el fondo a partir de los frames del
   buffer
22:   for row ← 0 to rows − 1 do
23:     for col ← 0 to cols − 1 do
24:       arrayPixeles ← ushort[tamanoBuffer]    ▷ Para cada pixel del fondo
25:       generamos un array con sus homologos en el buffer
26:       for i ← 0 to tamanoBuffer − 1 do
27:         arrayPixeles[i] ← buffer[i][row][col]
28:       end for
29:       fondo[row, col] ← FUNCIONAPRENDIZAJEFONDO(arrayPixeles)
30:     end for
31:   end for
32: end procedure
```

---

### *Técnicas de sustracción del fondo basado en aprendizaje por votación*

Para esta técnica se utiliza la base de los algoritmos que realizan el aprendizaje de estructuras votando por los candidatos, como por ejemplo se usa en el algoritmo de Hough (utilizado para la detección del ojo del pez en el primer caso práctico expuesto en esta tesis), en él el aprendizaje de los círculos en la imagen se realiza mediante una votación hecha por cada punto del borde a los posibles candidatos a círculos con distintos radios y diferente centro.

En esta aproximación, al igual que la aproximación de buffer, cada número determinado de frames ( $k$ ) se obtiene un frame de la escena y cada pixeles que compone

ese frame vota a los candidatos a fondo establecidos por (valor pixel  $\pm n$ ). Debido a que la variación que puede tener un pixel estable en el frame de profundidad es 3, se asigna ese valor a  $n$ . Cada pixel tendrá unos registros, que funcionarán como si fuesen urnas, que almacenan el valor, el número de votos de ese pixel y un bit que marcará si la urna está abierta o cerrada. Inicialmente estará todo vacío y en cuanto llegue el primer voto se añadirán diferentes urnas a la lista con los valores circundantes (una para cada valor), se añadirá también un voto y quedarán marcadas como abiertas. Según van ocurriendo nuevos votos se realizan comparaciones entre los valores circundantes del valor y las urnas de la lista, si la comparación encuentra una con la que coincida, se incrementará en uno sus votos y se marca como abierta. Si no coincide con ninguna, se creará una nueva urna.

Para el algoritmo se establece una tasa de refresco ( $N$ ) que va en correspondencia con el intervalo de captura de frames, estableciéndose finalmente que cada  $k*N$  frames se limpiarán las urnas cerradas y se cerrarán las abiertas. Además se realiza el recuento de los valores en las urnas y aquel valor más votado se establecerá como fondo. Si el valor más votado es el 0 (no alcanzable) se asignará el segundo valor más votado, siempre y cuando tenga por lo menos un 20% de los votos que tenía el 0.

Con este método los valores que se referencian frecuentemente durante largos periodos de tiempo mantendrán su grado de importancia aunque en cierto momento su frecuencia del fondo decrezca, debido a que persiste el número de votos. En caso de que durante los  $N$  frames anteriores al cierre de las urnas sus valores no sean referenciados, serán eliminados, permitiendo así que se pueda trabajar con fondos dinámicos.

El problema de las regiones intermitentes también queda solucionado con este método, ya que tanto el fondo real como el 0 (no alcanzable) persistirán a lo largo del tiempo consiguiendo un elevado número de votos y en el momento de clasificar el 0 como fondo podrá determinar si es una región no alcanzable o una intermitente consultando los votos del segundo valor más votado.

Sobre el algoritmo explicado se han realizado unas modificaciones para que agilicen su comportamiento pero manteniendo la filosofía del mismo. Estos cambios se realizan debido a que es un algoritmo computacionalmente complejo para su aplicación

en tiempo real. El principal cambio se hace en el sistema de votación, cuando un pixel nuevo es capturado, en vez de votar en todo el intervalo que lo rodea, vota solo por su valor. Además cuando introduce el voto en las urnas, a parte de su valor, representará  $n$  valores superiores e inferiores, de tal forma que si entra un valor nuevo que no cumple las restricciones ( $valor \pm n$ ) de las urnas que existen en ese momento, se crea una urna nueva, igual que se hacía en el algoritmo original, pero si el valor sí cumple las restricciones de alguna urna, esta será actualizada en su número de votos, su valor se cambiará a  $0.1*nuevoValor + 0.9*antiguoValor$  y será marcada como abierta.

Gracias a esta modificación se consigue agilizar el proceso, evitando guardar una urna para cada valor y además en el momento de realizar la votación solo se actualizará una única urna. A mayores es importante resaltar que este proceso se realiza para cada pixel del frame, así que es importante una buena optimización del proceso sin perder eficacia. En el caso explicado, se consiguen unos resultados tan buenos como en el original funcionando prácticamente al doble de velocidad.

---

**Algorithm 2** Votacion

---

```
1: intervaloCaptura ← 30
2: intervaloActualizacion ← intervaloCaptura * 30
3: ratioCeros ← 30
4: cols ← 640
5: rows ← 480
6: numFrame ← 0
7: fondo ← ushort[rows][cols]
8: frameUrnas ← lista < urna > [rows][cols]

9: procedure INTRODUCIRFRAME(frame)    ▷ Llamado cada vez que llega un nuevo
   frame
10:   numFrame ← numFrame + 1
11:   if (numFrame mód intervaloCaptura) = 0 then
12:     REALIZARVOTO(frame)
13:   end if
14:   if (frameNumber mód intervaloCaptura) = 0 then
15:     REALIZARRECUESTO(frame)
16:     ACTUALIZARFONDO()
17:   end if
18:   ...                                ▷ El procedimiento realiza más operaciones con el frame
19: end procedure

20: procedure REALIZARVOTO(frame)    ▷ Realiza la votación con los valores de un
   frame y sus circundantes
21:   for row ← 0 to rows - 1 do
22:     for col ← 0 to cols - 1 do
23:       valPixel ← frame[row][col]
24:       listaUrnas ← frameUrnas[row][col]
25:       if valPixel = 0 then          ▷ Si el valor es 0 solo se crea/actualiza su urna
26:         CREAMACTUALIZARURNA(valPixel, listaUrnas)
27:       else
28:         for n ← valPixel - (flexibilidad - 1) to valPixel + (flexibilidad - 1)
29:           ...                          ▷ Sino se crean/actualizan urnas en los valores circundantes
30:           CREAMACTUALIZARURNA(valPixel, listaUrnas)
31:         end for
32:       end if
33:     end for
34:   end for
35: end procedure
```

---

---

```

34: procedure CREARACTUALIZARURNA(valPixel, listaUrnas)
35:   existeUrna ← false
36:   for all u in listaUrnas do
37:     if u.valor = valPixel then                                     ▷ Actualizamos la urna
38:       u.votos ← u.votos + 1
39:       u.abierta ← true
40:       existeUrna ← true
41:     end if
42:   end for
43:   if existeUrna = false then                                       ▷ Creamos la urna porque no existe
44:     urna ← NUEVAURNA(valor = valPixel, votos = 1, abierta = true)
45:     listaUrnas.add(urna)
46:   end if
47: end procedure

48: procedure REALIZARRECUENTO(frame) ▷ Calcula el fondo a partir de los frames
del buffer
49:   for row ← 0 to rows - 1 do
50:     for col ← 0 to cols - 1 do
51:       listaUrnas ← frameUrnas[row][col]
52:       for all u in listaUrnas do
53:         if u.abierta = false then
54:           listaUrnas.delete(u)
55:         else
56:           u.abierta ← false
57:         end if
58:       end for
59:     end for
60:   end for
61: end procedure

62: procedure ACTUALIZARFONDO( ) ▷ Calcula el fondo a partir de los frames del buffer
63:   for row ← 0 to rows - 1 do
64:     for col ← 0 to cols - 1 do
65:       listaUrnas ← frameUrnas[row][col]
66:       urna ← ORDENARLISTA(listaUrnas, mayorQue) ▷ Ordenamos la lista
pasando la función que devuelve el mayor para que los coloque primero
67:       if listaUrnas.getByIndex(0) = 0 and listaUrnas.size() > 1 and
listaUrnas.getByIndex(1).votos > (listaUrnas.getByIndex(0)/ratioCeros) then
68:         fondo[row, col] ← listaUrnas.getByIndex(1)
69:       else
70:         fondo[row, col] ← listaUrnas.getByIndex(0)
71:       end if
72:     end for
73:   end for
74: end procedure

```

---

---

```

75: function MAYORQUE(urna1, urna2)      ▷ Funcion de ordenación que devuelve la
    mayor de las dos urnas considerando el 0 como el valor máximo
76:   if urna1.votos > urna2.votos then
77:     u ← urna1
78:   else
79:     u ← urna2
80:   end if
81:   return u
82: end function

```

---

Una vez aplicado el algoritmo para descartar el fondo, el siguiente paso es utilizar la técnica de máscara para obtener los objetos en movimiento (que serán candidatos a detectarse como pez). Dado que se trabaja con una imagen que representa profundidad, se restará el frame de fondo y se mantendrán los pixeles que superen un umbral establecido (después de diversas pruebas este umbral se ha establecido en 15).

---

**Algorithm 3** CalcularMascara

---

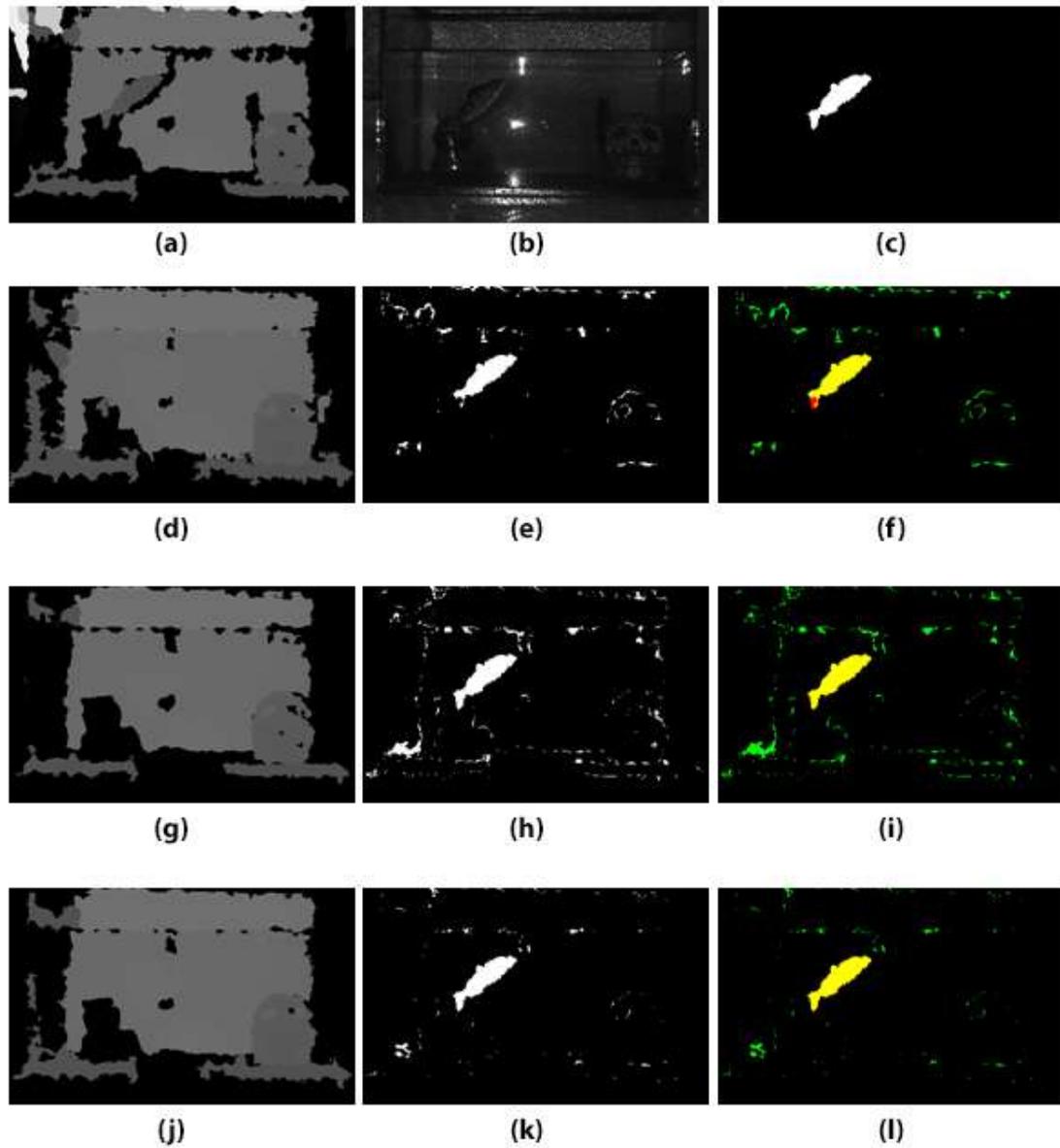
```

1: cols ← 640
2: rows ← 480
3: umbral ← 10
4: function CALCULARMASCARA(frame)
5:   for row ← 0 to rows - 1 do
6:     for col ← 0 to cols - 1 do
7:       diferencia ← frame[row][col] - fondo[row][col]
8:       if (frame[row][col] ≠ 0) and
9:         (fondo[row][col] = 0 or diferencia < -umbral) then
10:        mascara[row][col] ← 255
11:      else
12:        mascara[row][col] ← 0
13:      end if
14:    end for
15:  end for
16:  return mascara
17: end function

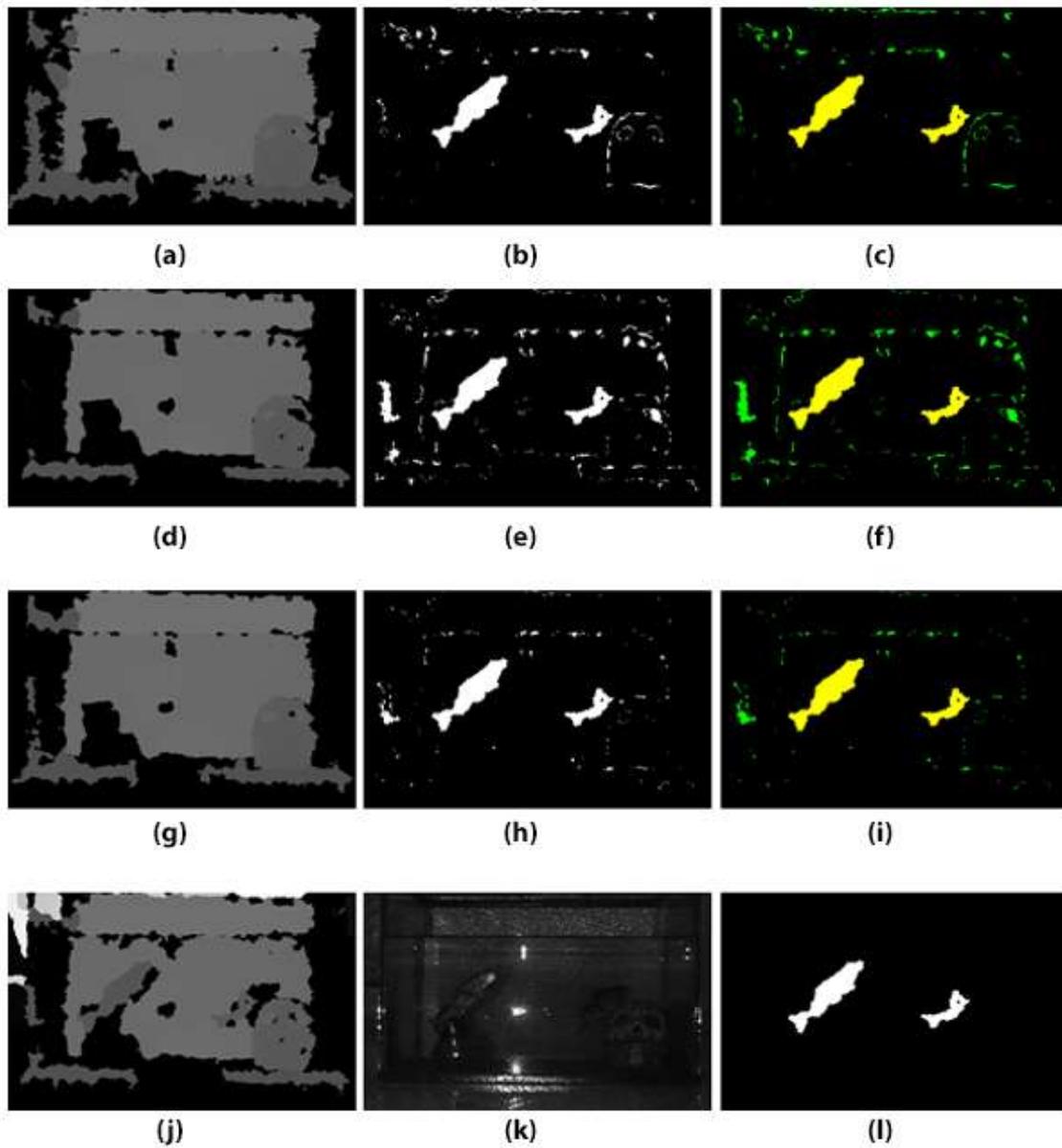
```

---

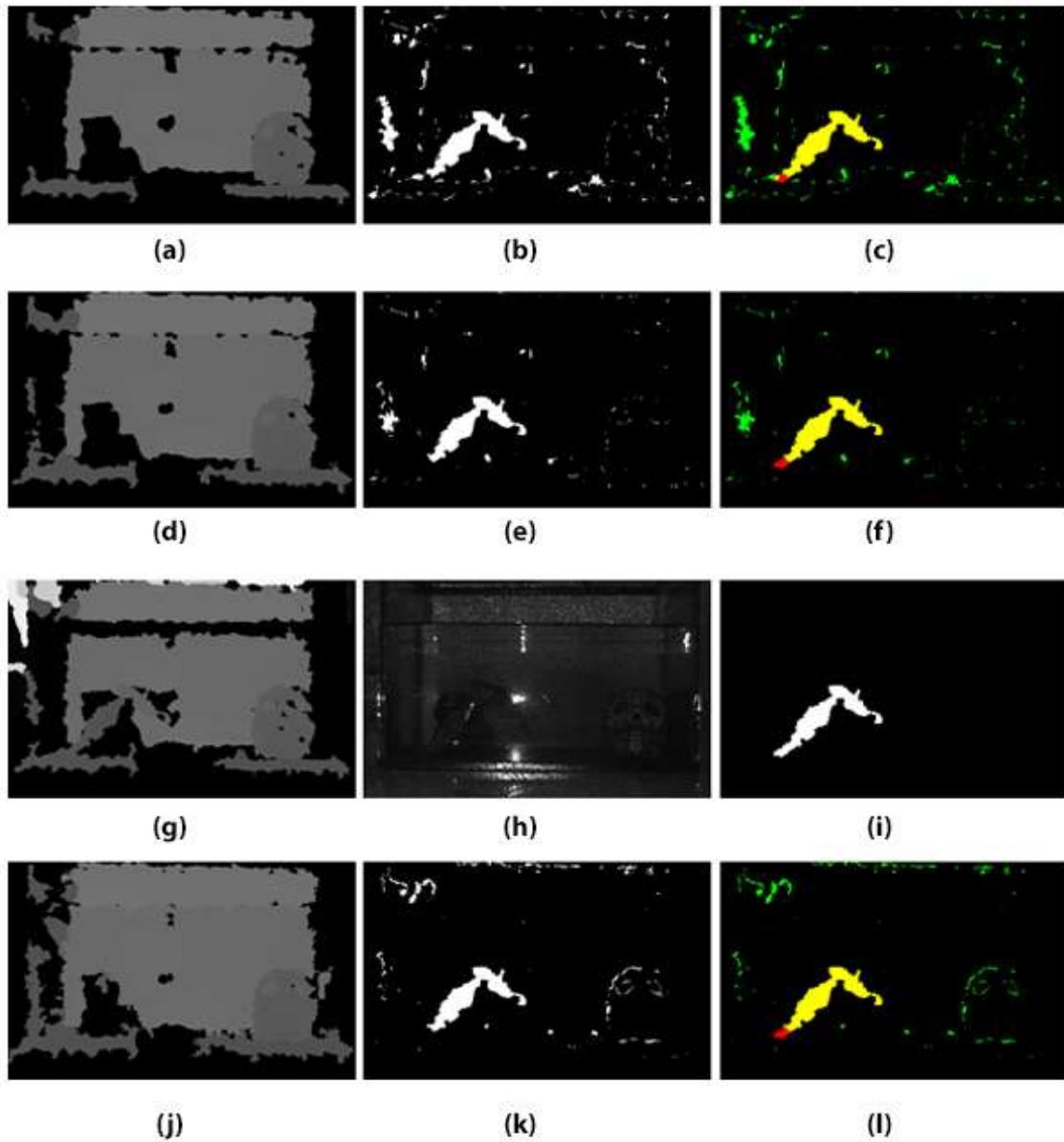
Para comprobar el correcto funcionamiento de estos algoritmos, se han generado varias pruebas de funcionamiento en entorno real, utilizando peces artificiales simulando diferentes condiciones que puedan darse.



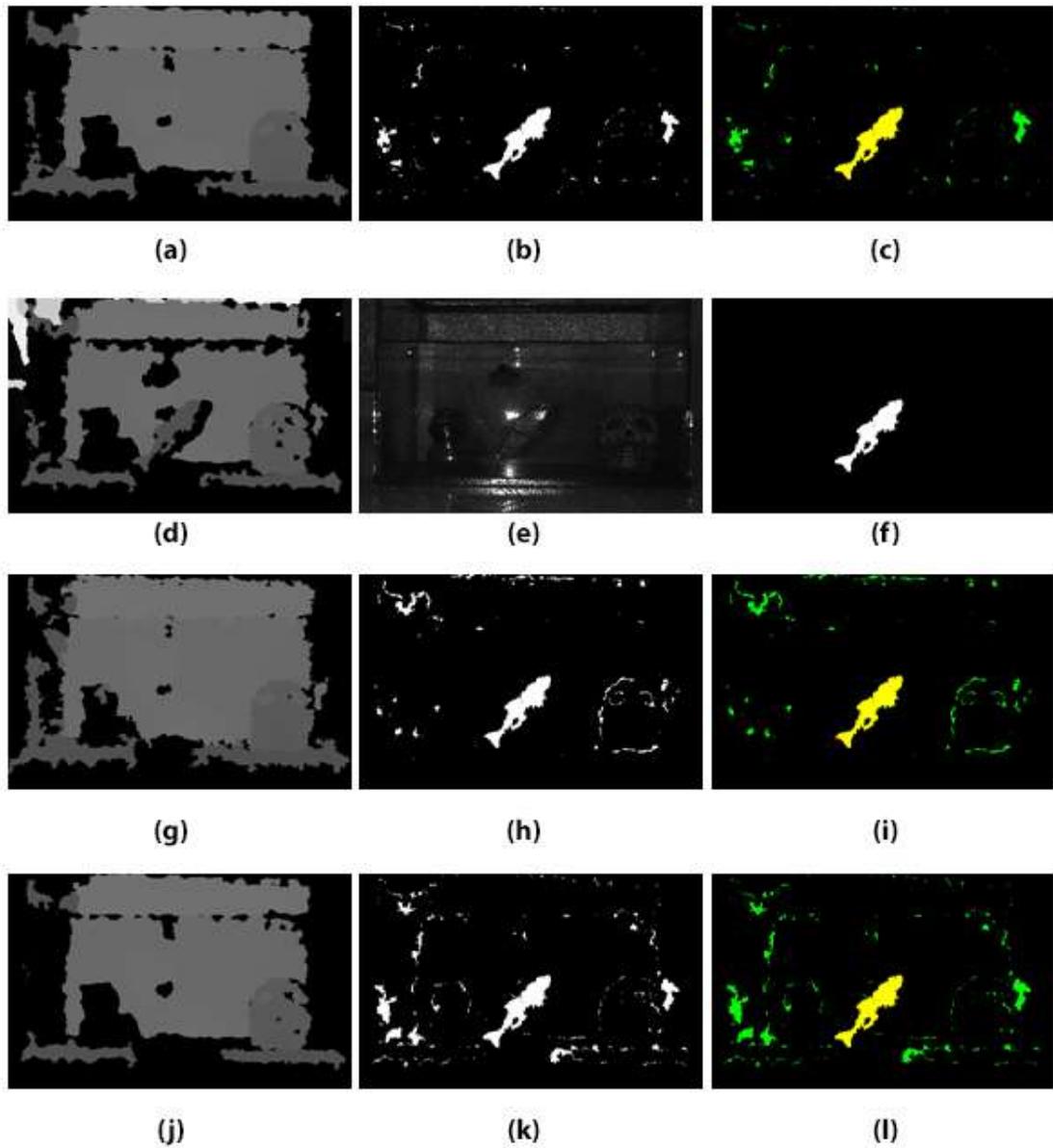
**Figura 57 :** Prueba con un pez que se identifica correctamente (en amarillo): (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscara máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscara mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación



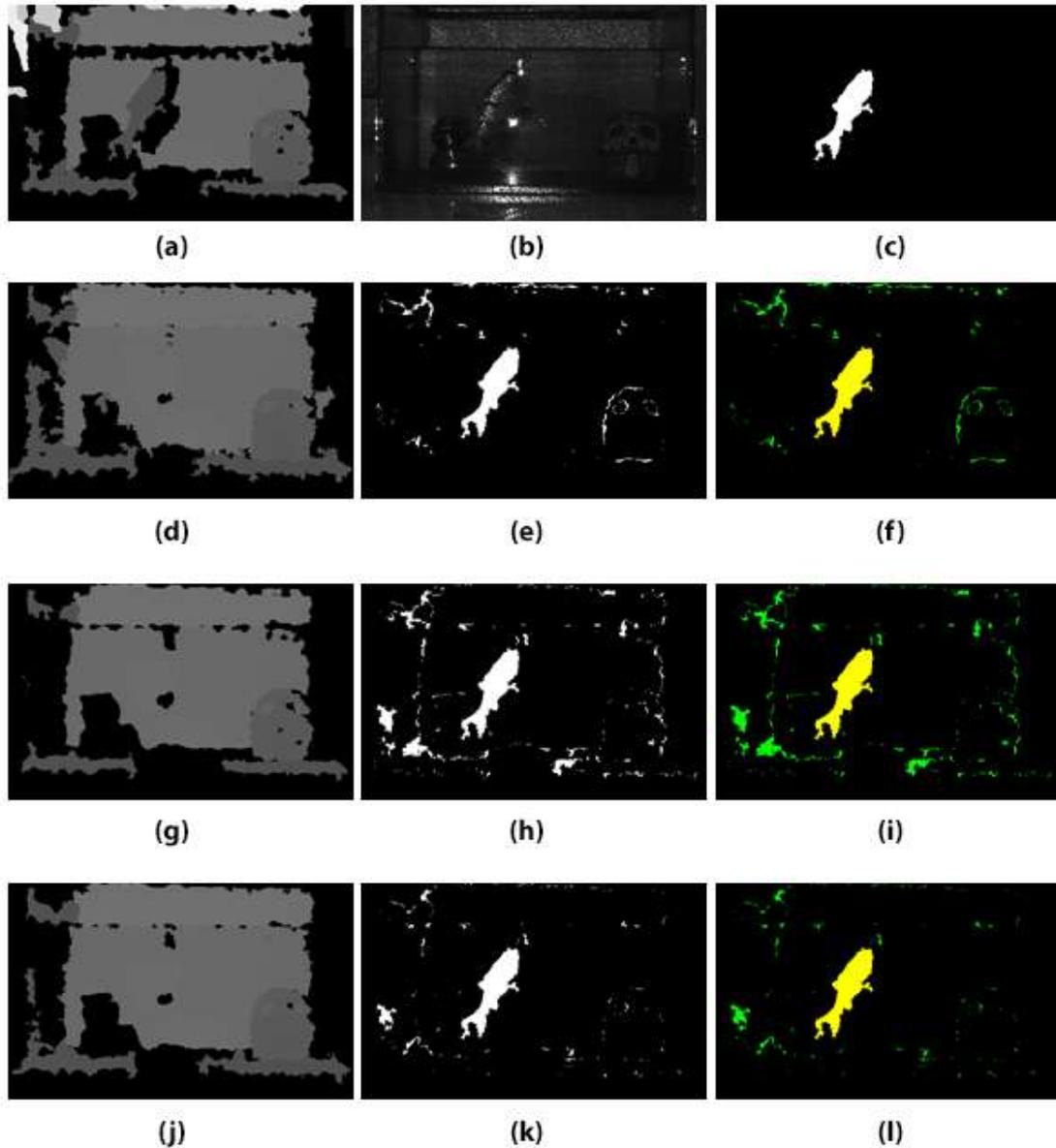
**Figura 58:** Prueba con dos peces de diferente tamaño (identificados en amarillo) : (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscara máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscara mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación



**Figura 59:** Prueba con 2 peces que se solapan: (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscara máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscara mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación



**Figura 60:** Prueba con un pez de diferente forma y tamaño: (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscara máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscara mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación



**Figura 61:** Prueba con un pez de diferente forma y tamaño: (a) profundidad, (b) infrarrojo, (c) ground truth, (d) fondo máximo, (e) máscara máximo, (f) diferencia máximo, (g) fondo mediana, (h) máscara mediana, (i) diferencia mediana, (j) fondo votación, (k) máscara votación, (l) diferencia votación

## 5. Recepción de la información en tiempo real

Combinando los frames infrarrojo y de profundidad se obtiene un nuevo frame. Como el frame de profundidad se calcula a partir del de infrarrojos, el punto  $(x,y)$  de ambos frames representa el mismo punto. Por lo tanto se puede realizar operaciones con el nuevo frame superponiendo ambos frames o bien obtener un nuevo frame con dos canales.

Para la combinación del frame de color con el de profundidad hay que hacer uso del SDK de Kinect para a través de una matriz de correspondencia, emparejar ambos frames. Esto es debido a que las cámaras que obtienen dichos frames están desplazadas, por lo que en un principio el punto  $(x,y)$  de un frame no se corresponde con el  $(x,y)$  del otro frame.

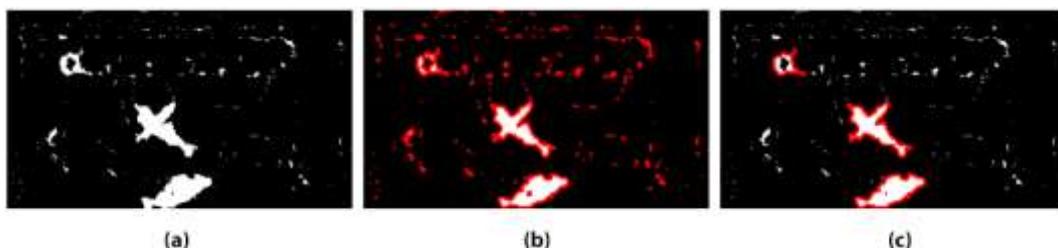
Teniendo estas combinaciones se podrá utilizar uno o varios frames para cada etapa. Por ejemplo, como se vio anteriormente el frame de infrarrojos no se utiliza para la substracción del fondo, ya que con el de profundidad es suficiente para conseguir unos buenos resultados.

## 6. Detección de objetos

### *Segmentación del pez.*

El siguiente paso es la detección del pez y su contorno. Dado que el objetivo del sistema es la detección de múltiples peces en una misma imagen, pueden darse múltiples situaciones. Uno de los casos más complejos es donde un pez está delante de otro y es necesario que el sistema identifique correctamente a ambos.

Lo primero que se desea obtener son todas las regiones de movimiento, filtrando las que son demasiado pequeñas. Se utiliza la función de *OpenCV findContours* aplicándola a la máscara de entrada y posteriormente se realiza un filtrado de aquellos contornos cuya área sea inferior a 100 píxeles (Figura 62).



**Figura 62:** Detección de regiones con movimiento (a) máscara, (b) contornos de la máscara, (c) contornos válidos

Para cada una de las regiones de movimiento se obtiene el marco mínimo que contiene a cada uno de los contornos, obteniendo a partir de él la imagen de profundidad de la región recortada en el frame de profundidad original (Figura 63).



**Figura 63:** Región con movimiento (a) Identificación de la región, (b) Profundidad de la región filtrada

El siguiente paso es separar los objetos que aparecen en el interior de las regiones, para ello, sobre la imagen de profundidad, se utiliza el detector de bordes *Canny* [74]. Adicionalmente, al detectar los bordes, se puede detectar zonas donde los bordes no están conectados, para ello se realiza un sistema automático de conexión de bordes utilizando las direcciones más posibles para su conexión, basándose en los puntos detectados como no conexos (Figura 64).



**Figura 64:** Detección de bordes (a) máscara de bordes, (b) puntos seleccionados para la unión de bordes

El siguiente paso es la extracción de siluetas para poder identificar los objetos candidatos a ser pez. Para ello se identifican aquellas regiones cerradas con bordes (Figura 65 y Figura 66).



**Figura 65:** Siluetas candidatas que pueden representar uno o dos peces

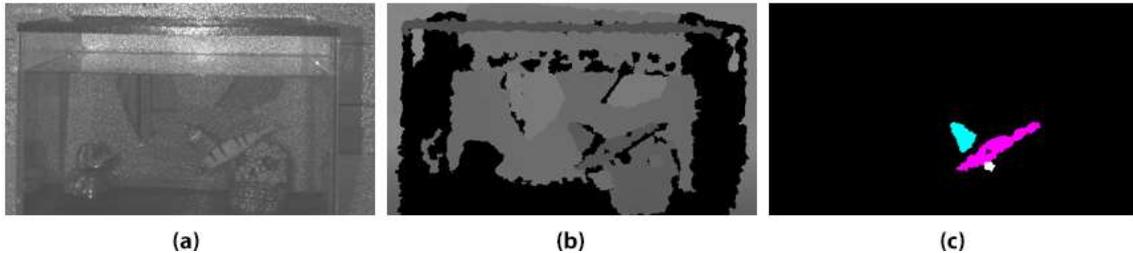


**Figura 66:** Prueba de detección de siluetas con peces de 3.5 cm de longitud

El siguiente paso que debe realizar el sistema es la obtención de características de las siluetas. Se va a hacer uso de las imágenes de las siluetas obtenidas en la fase de segmentación, es decir la imagen de profundidad, su máscara y su contorno. Con ellas se puede realizar una extracción de características que permita evaluar posteriormente si el objeto es un pez, una parte de un pez que sea necesario combinarla con otra o si es otro objeto y no un pez (Figura 67). El algoritmo de manera automática obtiene las siguientes características:

- La elipse que minimiza el error de los puntos del contorno.
- El área, en píxeles, que ocupa el contorno
- Los extremos de la silueta en el espacio real

- Longitud y ancho de la silueta
- La posición en el espacio real
- La envolvente conexa y su área

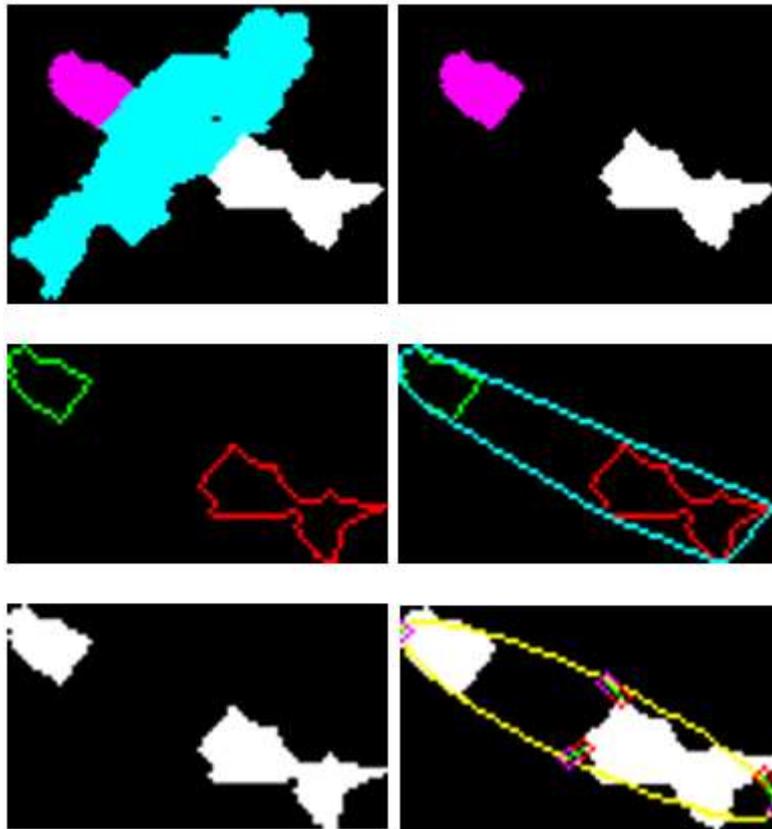


**Figura 67:** Detección y extracción de siluetas (a) infrarrojo, (b) profundidad, (c) siluetas extraídas

Con estas características se desarrolla un algoritmo que clasifica si un objeto detectado es un pez o no. El clasificador utilizado se trata de un SVM (*Support Vector Machine*) [89], un algoritmo de aprendizaje supervisado que a partir de una serie de características obtenidas de ejemplos positivos y negativos aprende una función matemática que es capaz de discernir si un objeto pertenece al grupo positivo o al negativo, es decir, si es pez o no. Dado que el objetivo es eliminar lo máximo posible los falsos positivos, se ha entrenado este algoritmo con exigencias en la detección. Dado que se dispone de 30 imágenes por segundo, el hecho de que el clasificador no detecte el pez en una o varias imágenes no supone ningún problema pues luego se utilizará un algoritmo de seguimiento que permitirá seguir la silueta del pez. En cambio, si se producen falsos positivos, el algoritmo de seguimiento no funcionará adecuadamente y la detección de un pez cuando no lo es supone un problema mayor.

En cada región clasificada de potencial candidatos a pez, también es necesario aplicar una técnica de combinación de siluetas. A la hora de realizar la combinación de una silueta con otra se busca obtener una nueva silueta que represente la hipotética silueta ocluida, para ello es necesario fusionar las características de las dos siluetas individuales y esto no se trata de un proceso trivial, porque ambas siluetas no se encuentran conectadas y (en el caso de tratarse de un pez ocluido) el hecho de no conocer una parte del hipotético objeto limita la capacidad de medición del mismo.

El primer paso de la combinación consiste en el establecimiento de un nuevo marco para la silueta combinada, un marco que englobe los marcos de cada una de las siluetas individuales, posteriormente se procede a la obtención del contorno de la silueta combinada a partir de los contornos individuales que son almacenados en la nueva silueta, luego se obtiene la envolvente convexa de la combinación, que es la envolvente convexa de ambos contornos.



**Figura 68:** Proceso de fusión de partes de siluetas y detección del pez ocluido

A la hora de evaluar si dos siluetas menores deben ser reconstruidas se ha diseñado un método basado en la evaluación de la distancia a la que se encuentran los puntos extremos de los objetos de la elipse del resto de objetos de la escena. En el caso de encontrarse muy cerca, el sistema ha de prepararse para una futura oclusión, por lo que se realiza la búsqueda de una silueta pequeña que tenga alguno de sus puntos extremos, pegado a la elipse del objeto más cercano, y en el caso de que cumpla ciertos requisitos se realiza la reconstrucción con la silueta ocluida y la nueva silueta (Figura 68).

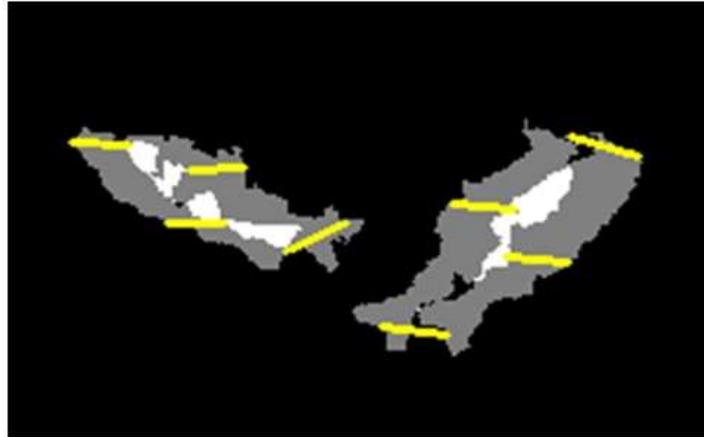
## 7. Seguimiento de objetos

En esta fase se trata de mantener la continuidad de la información sobre los objetos detectados en las fases anteriores. En esas fases no se tienen en cuenta las detecciones pasadas y futuras, pero es interesante tener en cuenta esta información cuando la aplicación dispone de la posibilidad de recibir varios frames a lo largo del tiempo, para mantener una coherencia y persistencia en las detecciones que se realizan en frames sucesivos.

El método implementado para realizar el seguimiento se basa en emparejar los objetos detectados en el frame  $N$  con los que se encuentran más cercanos a ellos en el siguiente frame. Aprovechando que se conoce la posición en el mundo real, para realizar el emparejamiento, se da prioridad a los objetos que se encuentran más cerca de la Kinect.

Cuando se recibe el primer frame, por lo que no hay que realizar emparejamiento ninguno, se dispone de los peces detectados en una fase anterior y se almacenan las características de la detección: posición, los 4 puntos extremos de la silueta, la altura y la anchura. Además se asigna un color aleatorio y un contador numérico para identificarlos en la imagen que se mostrará al usuario. Todos estos datos forman una firma de esa detección.

Posteriormente llegarán las detecciones del siguiente frame y se ordenan la lista de firmas de las detecciones poniendo en primer lugar las que se posicionan más cerca de la cámara. Una vez ordenadas, se busca la detección que minimice la suma de las distancias de sus puntos extremos (Figura 69).

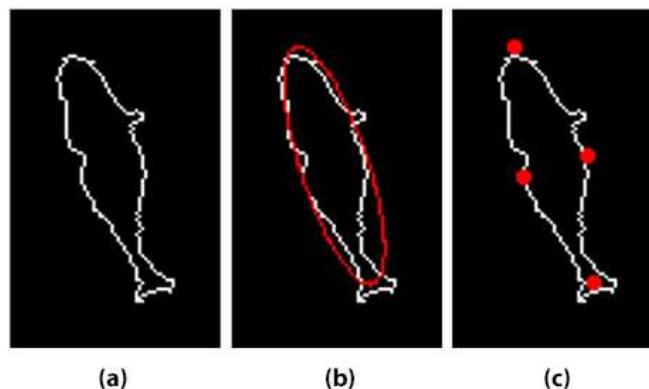


**Figura 69:** Seguimiento de dos siluetas en frames consecutivos por medio de la minimización de la distancia de sus puntos extremos (en amarillo)

## 8. Obtención del tamaño del pez

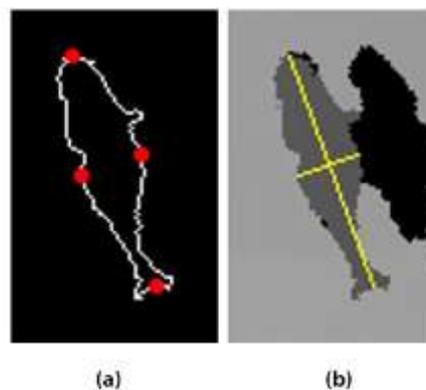
Para llevar a cabo la medición automática de un objeto es necesaria la obtención automática de los extremos del mismo a partir de los contornos de las siluetas obtenidos en fases anteriores.

Para el diseño de este algoritmo, al igual que en las otras partes de esta tesis, se utiliza la librería de visión artificial *OpenCV*. Sobre el contorno detectado, usando la función *fitEllipse*, se busca la elipse que haga mínima la distancia a los puntos del contorno (Figura 70).



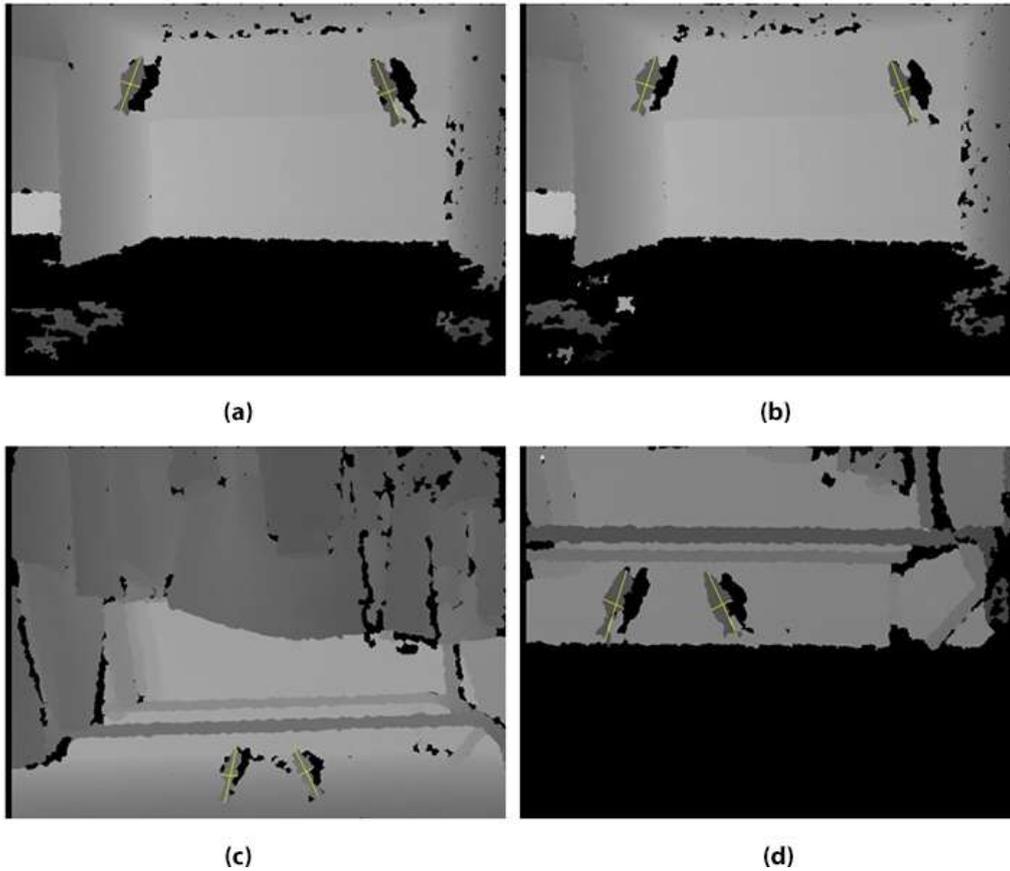
**Figura 70:** Extracción de elipse (a) contorno, (b) elipse, (c) extremos de la elipse

A continuación se calculan la posición en el espacio real de los 4 puntos extremos de cada eje de la elipse. Para ello hay que localizar los puntos del contorno que estén más cerca de los puntos extremos de cada eje de la elipse (Figura 70c). Estos serán los puntos extremos del objeto y obteniendo su valor en el frame de profundidad se saben las distancias de esos 4 puntos al plano de la Kinect. Con esta información y la posición de los puntos en el frame de profundidad se utiliza la función *NuiImageTransform* para calcular las coordenadas de los puntos extremos con respecto a la Kinect en el espacio real. Con esas posiciones reales, se calcula la distancia euclídea entre los puntos para obtener la longitud y anchura real del objeto (Figura 71).



**Figura 71:** A partir de los puntos extremos (a) se calculan las distancias euclídeas (b)

Una vez desarrollado este algoritmo se han realizado un conjunto de ensayos para comprobar el correcto funcionamiento y la visualización de la detección de los objetos en situaciones pseudo-reales, donde se utilizan peces artificiales en una pecera llena de agua (Figura 72).



**Figura 72:** Frames utilizados en las pruebas con puntos extraídos automáticamente (marcados en amarillo)  
(a) frame 1, (b) frame 2, (c) frame 3, (d) frame 4

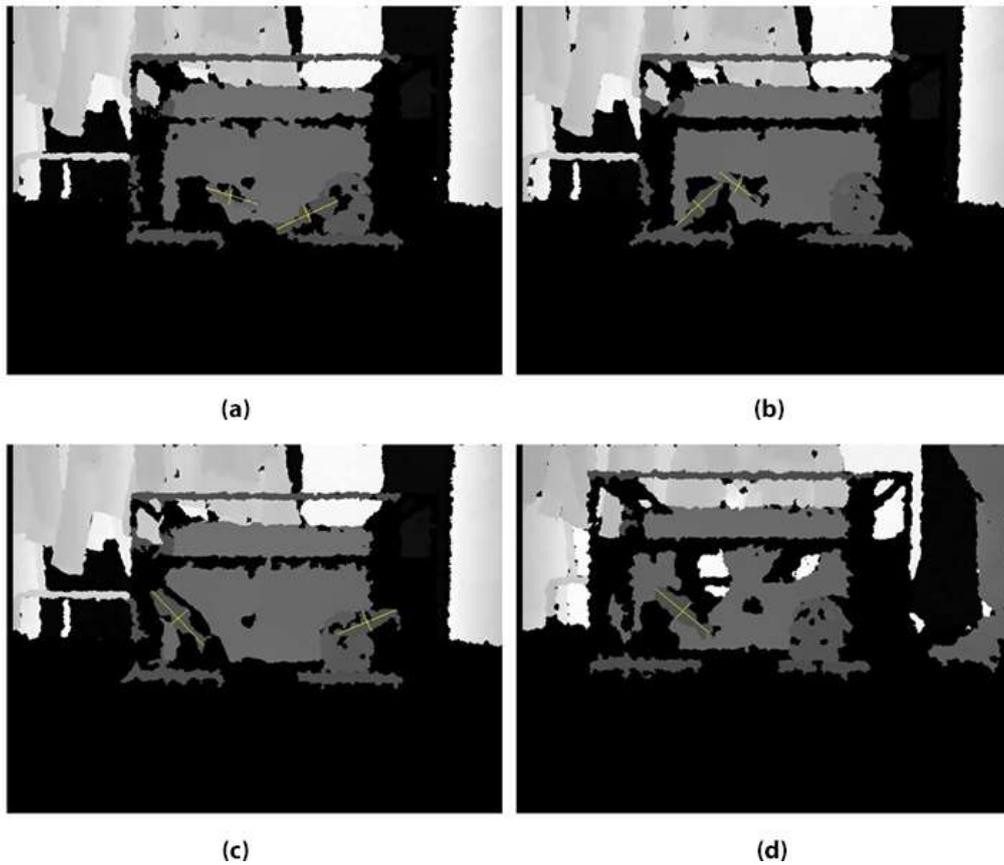
<i>Frame</i>	<i>Posición (x,y,z) (mm)</i>	<i>Tamaño Real (largo x ancho) (mm)</i>	<i>Medición automática</i>
<i>Figura 72(a)</i>	(228,105,702)	108x28	(228,105,702)
<i>Figura 72(a)</i>	(-200,235,781)	101x32	(-200,235,781)
<i>Figura 72 (b)</i>	(220,106,689)	105x24	(220,106,689)
<i>Figura 72 (b)</i>	(-199,231,774)	98x32	(-199,231,774)
<i>Figura 72 (c)</i>	(127,-336,921)	103x30	(127,-336,921)
<i>Figura 72 (c)</i>	(-40,-238,916)	110x38	(-40,-238,916)
<i>Figura 72 (d)</i>	(-50,-3,677)	105x32	(-50,-3,677)

**Tabla 4:** Resumen de los resultados en los ensayos

Las mediciones de objetos en entornos normales sirven para asegurar que se está realizando una correcta medición automática, pero en el entorno propuesto existen más variables que hacen que las mediciones puedan ser erróneas (agua, cristal, luminosidad y

suciedad tanto del agua como del cristal) y estas mediciones deben de ser corregidas y probadas de alguna manera. Se han realizado pruebas similares a las anteriores (

Figura 73) con los peces dentro de la pecera y conociendo la distancia real existente. En estas pruebas se realizará una medición corregida mediante la aplicación de la formula obtenida previamente, pasando como entrada al algoritmo de medición la distancia del sensor a la pecera.



**Figura 73:** Frames utilizados en las pruebas con puntos extraídos automáticamente (a) frame 1, (b) frame 2, (c) frame 3, (d) frame 4

<i>Frame</i>	<i>Posición (x,y,z) (mm)</i>	<i>Distancia a pecera</i>	<i>Tamaño Real (largo- Xancho) (mm)</i>	<i>Medición estandar</i>	<i>Medición aplicando corrección</i>
<i>Figura 73(a)</i>	(121,-14,650)	650	110x30	99x27	100x28
<i>Figura 73(a)</i>	(4,-24,832)	650	100x32	96x28	104x31
<i>Figura 73(b)</i>	(-64,10,657)	647	110x30	101x29	106x32
<i>Figura 73(b)</i>	(-14,-17,836)	647	100x32	82x27	87x29
<i>Figura 73(c)</i>	(206,28,647)	647	100x32	94x32	99x33
<i>Figura 73(c)</i>	(-78,-24,678)	647	110x30	114x34	115x34
<i>Figura 73(d)</i>	(-81,-4,648)	613	100x32	100x34	102x35

**Tabla 5:** Resumen de los resultados en los ensayos

El método utilizado a la hora de medir los peces arroja resultados satisfactorios, la corrección utilizada reduce el error en las mediciones por lo que es recomendable implantarla.

## 9. Visualización en tiempo real

Una vez desarrollado el algoritmo, para facilitar el uso se desarrolla una interface gráfica, también incluida en el ámbito del TFG desarrollado. Esta interface se ha desarrollado como una única ventana donde se obtienen los diferentes videos obtenidos de la Kinect, y mediante un menú con varias opciones se pueden establecer diferentes configuraciones de funcionamiento.



**Figura 74:** Ejemplo de pantalla del software de visualización en funcionamiento.

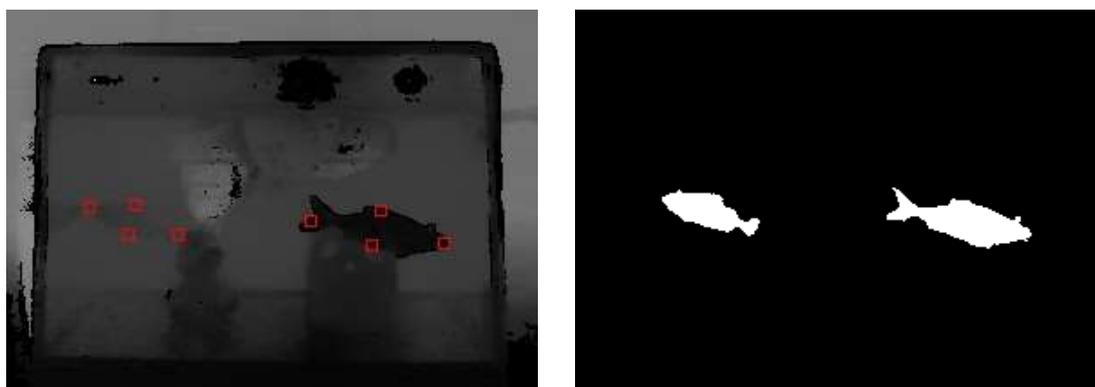


**Figura 75:** Ejemplo de pantalla con la detección de 2 peces y la obtención de su tamaño.

Tal como puede observarse en las pantallas previas (Figura 74 y Figura 75), el software aplica el algoritmo desarrollado, y sobre las imágenes de profundidad obtenidas de la Kinect dibuja el contorno del pez detectado en diferentes colores para cada pez, y a la derecha de la pantalla se muestra una lista de números representando a cada pez con su color y las características de posición y tamaño (alto y largo).

### 7.3. Resultados experimentales

Una vez desarrollado completamente el algoritmo, es el momento de probarlos en diferentes situaciones, primero sus partes individualmente y posteriormente en su conjunto, tanto con peces artificiales simulando las posibles combinaciones de situaciones que se pueden dar y posteriormente con peces reales en su entorno. Para este último caso también se ha contado con la ayuda del Aquarium Finisterrae de la ciudad de A Coruña, que cuenta con varios tanques de peces de diferentes tamaños y entornos de iluminación y visibilidad.



**Figura 76:** Ejemplo de detección de dos peces de diferente tamaño en condiciones de baja luminosidad

Dado que la Kinect se basa en la emisión de luz infrarroja, la ausencia de luz visible no supone un problema a la hora de realizar la detección de objetos, como se observa en la Figura 76.

Posteriormente se ha realizado un conjunto de pruebas con peces artificiales para comprobar su correcta detección y tamaño. La Tabla 6 muestra el resumen de estas pruebas.

<i>Prueba</i>	<i>Número de frames</i>	<i>Medida Real</i>	<i>Medición del sistema</i>
1	43	110x30	111x24
2	29	110x30	111x29
3	33	110x30	113x31
4	49	110x30	112x31
5	46	110x30	114x31
6	29	110x30	111x30
7	60	100x32	102x28

**Tabla 6:** Resumen de los resultados en los ensayos

Una vez comprobado el correcto funcionamiento del sistema se pasa a pruebas en entorno real. Para ello, se utilizan dos tanques de peces existentes en el Aquarium Finisterrae de A Coruña. A continuación, se muestran unas fotografías de algunos de los tanques (Figura 77, Figura 78, Figura 79 y Figura 80).



**Figura 77 :** Imagen de uno de los tanques del Aquarium Finisterrae



**Figura 78 :** Imagen de uno de los tanques del Aquarium Finisterrae



**Figura 79 :** Imagen de uno de los tanques del Aquarium Finisterrae



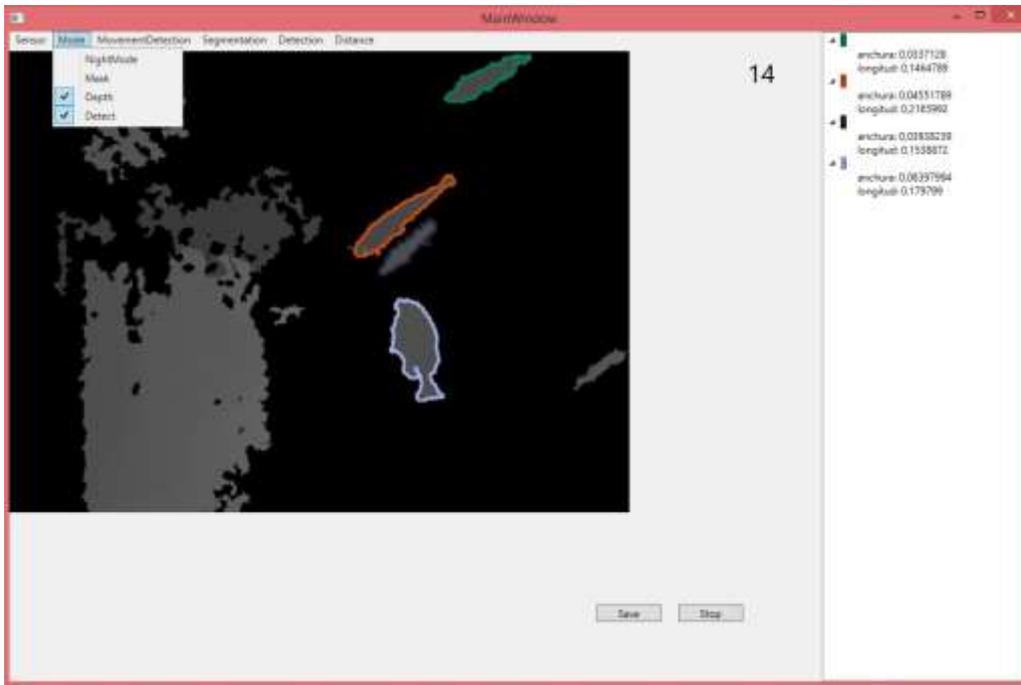
**Figura 80:** Imagen de uno de los tanques del Aquarium Finisterrae

A continuación, se muestran varias capturas de pantallas (Figura 81, Figura 82, Figura 83 y Figura 84) de funcionamiento del software desarrollado capturando en tiempo real las imágenes de infrarrojo y profundidad de la Kinect y procesadas en tiempo real para visualizarlas en el software donde se detectan los peces y se muestran sus características de tamaño. La unidad de distancia utilizada es el metro, que son los valores que se visualizan en el software para la longitud y anchura del pez.

Es importante indicar que debido al cambio de entorno y al aumento de las dimensiones de los peces que se encuentran en estos tanques (más del doble que en entorno no real), implica un ajuste en los parámetros del algoritmo y, como no está entrenado para este entorno, se desactiva la detección mediante el SVM.

En estos ensayos no se conocen las medidas reales de los peces, por lo que no se pueden comparar con las calculadas por el sistema. Sin embargo, sí que se aprecia una correlación entre las medias de cada pez en los diferentes frames que puede indicar que la medición está en sintonía con las pruebas en entorno no real, en donde las mediciones

se aproximaban en gran término a la realidad. Este es un aspecto que se podrá verificar en pruebas futuras.



**Figura 81:** Ejemplo de detección en la herramienta en un tanque con peces reales



**Figura 82:** Ejemplo de detección en la herramienta en un tanque con peces reales



Figura 83: Ejemplo de detección en la herramienta en un tanque con peces reales



Figura 84: Ejemplo de detección en la herramienta en un tanque con peces reales

## 7.4. Conclusiones

En este caso práctico se utilizan los sensores de 3D para el desarrollo de una técnica de detección y estimación del tamaño de peces. Como en los casos anteriores, se ha combinado su uso con técnicas de visión artificial para lograr una detección y estimación en tiempo real.

En cuanto al sensor utilizado, la Kinect no deja de ser un dispositivo pensado para el mundo de los videojuegos y para ser asequible para su comercialización. Por lo tanto, las prestaciones no serán las mismas que puedan conseguirse con un sensor específico que hará mejorar los resultados. Una de las limitaciones que tiene es que la visión y el rango están limitados, a pesar de ello cuando realiza las mediciones estas son muy precisas.

Otra de las limitaciones de la Kinect por lo observado en este caso, es que se recomienda situar el sensor a una distancia aproximada de 65 cm del cristal y los peces no pasen a más de 40 cm del cristal. Cumpliendo estos requisitos y configurando la herramienta con estas distancias, los peces serían detectados por el sensor y medidos por el algoritmo.

Los resultados en entorno no real muestran unas estimaciones de tamaño correctas, sin embargo en entorno real, como se comentó, no se disponen de las medidas reales para comparar con las estimaciones de la técnica propuesta. En el futuro se debe de intentar realizar experimentos en situaciones en donde se conozcan las medidas reales, para llevar a cabo una validación completa de la técnica.

## 8. Conclusiones y trabajo futuro

### 8.1. Castellano

En esta tesis se han generado tres técnicas que consiguen validar la hipótesis de realizar la detección del pez en el agua sin que se produzca intrusión ni contacto físico con el pez gracias a la teledetección. Mediante el uso de una metodología común, la información adquirida en cada uno de los casos es tratada con técnicas de visión artificial y utilizando diferentes algoritmos se llega a una confirmación o rechazo de si esa detección de pez es correcta o no, para posteriormente realizar una estimación de su tamaño.

En el primer caso práctico se utilizan dos cámaras de vídeo para generar una imagen con información en 3 dimensiones que permite extraer información sobre el objeto y la distancia a la que se encuentra de las cámaras. Para validar si el objeto detectado es un pez o no se busca en la imagen el ojo del pez en un primer escenario utilizando la detección de Hough y en un segundo mediante el uso de redes de neuronas artificiales para actuar cuando el primer algoritmo no sea capaz de hallar el ojo.

En el segundo caso práctico se complementa el uso de una cámara con el de un sensor infrarrojo. Mediante el sensor infrarrojo se detecta el objeto que interfiere el haz emitido (informa de la distancia a la que se encuentra el objeto) y en ese instante se toma una imagen del objeto que es analizada usando la visión artificial para decidir si es un pez o no.

En el último caso se hace uso de un sensor de luz estructurada que genera una nube de puntos con información del escenario escaneado. Con esa información combinada con los diferentes tipos de frames que aporta el dispositivo utilizado, se aplican las técnicas de visión artificial que permiten realizar la detección y medición de los peces.

Cada caso ha sido testado primero en laboratorio y posteriormente en un entorno real, aunque ligeramente controlado. En el futuro sería interesante dar un paso más en la investigación y poder realizar ensayos en situaciones aún más cercanas a la realidad y que permitirían ajustar y depurar la técnica. Uno de los problemas que han surgido en las pruebas realizadas, y que debe estudiarse cuál sería la mejor manera de solventar, es poder disponer de datos concretos de los peces que se monitorizan para afinar al máximo la técnica en su apartado de medición. Sin embargo, se es consciente de cuánto más cercano a la realidad sea el entorno de prueba, más complicado será tener esta información. En otros casos se disponía de esa información pero era complicado hacer la correlación de los especímenes en las imágenes debido al número de ejemplares disponibles.

Después de probar y analizar los resultados en los tres casos se generan las siguientes conclusiones:

- El caso práctico 1 con las cámaras de visión estéreo y el uso de redes de neuronas artificiales ralentiza su utilización en tiempo real más que en el caso práctico 2. También se debe tener en cuenta que el cambio en las condiciones sobre las que se entrenó la red o la inclusión de nuevas especies conllevarían el reentrenamiento de la red de neuronas artificiales.
- El caso práctico 3 con el dispositivo Kinect, tiene limitaciones en cuanto a los rangos de visión y en que se tiene que situar el dispositivo a 65 cm del cristal y los peces no pueden pasar a más de 40 cm del cristal.
- El caso práctico 2 parece ser el que mejor combina el uso en tiempo real con la detección y medición de los peces. Habría que tener en cuenta que la inclusión de nuevas especies haría necesario también la introducción de nuevos patrones de esa especie que permitan realizar una correcta clasificación del objeto detectado por el láser, como pez.

## 8.2. Galego

Nesta tese xeráronse tres técnicas que conseguen validar a hipótese de realizar a detección do peixe na auga sen que se produza intrusión nin contacto físico co peixe grazas á teledetección. Mediante o uso dunha metodoloxía común, a información adquirida en cada un dos casos é tratada con técnicas de visión artificial e utilizando diferentes algoritmos chégase a unha confirmación ou rexeitamento de se esa detección de peixe é correcta ou non, para posteriormente realizar unha estimación do seu tamaño.

No primeiro caso práctico utilízanse dúas cámaras de vídeo para xerar unha imaxe con información en 3 dimensións que permite extraer información sobre o obxecto e a distancia á que se atopa das cámaras. Para validar se o obxecto detectado é un peixe ou non se busca na imaxe o ollo do peixe nun primeiro escenario utilizando a detección de Hough e nun segundo mediante o uso de redes de neuronas artificiais para actuar cando o primeiro algoritmo non sexa capaz de atopar o ollo.

No segundo caso práctico complementábase o uso dunha cámara cun sensor infravermello. Mediante o sensor infravermello detéctase o obxecto que interfere o feixe emitido (informa da distancia á que se atopa o obxecto) e nese instante tómase unha imaxe do obxecto que é analizada usando a visión artificial para decidir se é un peixe ou non.

No último caso faise uso dun sensor de luz estruturada que xera unha nube de puntos con información do escenario escaneado. Con esa información combinada cos diferentes tipos de frames que achega o dispositivo utilizado, aplícanse as técnicas de visión artificial que permiten realizar a detección e medición dos peixes.

Cada caso foi probado primeiro en laboratorio e posteriormente nunha contorna real, aínda que lixeiramente controlado. No futuro sería interesante dar un paso máis na investigación e poder realizar ensaios en situacións aínda máis próximas á realidade e que permitirían axustar e depurar a técnica. Un dos problemas que xurdiron nas probas realizadas, e que debe estudarse cal sería a mellor maneira de liquidar, é poder dispoñer de datos concretos dos peixes que se monitorizan para afinar ao máximo a técnica no seu apartado de medición. Con todo, sábese que canto máis próximo á realidade sexa a

contorna de proba, máis complicado será ter esta información. Noutros casos dispoñíase desa información pero era complicado facer a correlación dos espécimes nas imaxes debido ao número de exemplares dispoñibles.

Despois de probar e analizar os resultados nos tres casos xéranse as seguintes conclusións:

- O caso práctico 1 coas cámaras de visión estéreo e o uso de redes de neuronas artificiais retarda a súa utilización en tempo real máis que no caso práctico 2. Tamén se debe de ter en conta que o cambio nas condicións sobre as que se adestrou a rede ou a inclusión de novas especies conlevarían repetir o proceso de adestramento da rede de neuronas artificiais.
- O caso práctico 3 co dispositivo Kinect, ten limitacións en canto aos rangos de visión e en que se ten que situar o dispositivo a 65 cm do cristal e os peixes non poden pasar a máis de 40 cm do cristal.
- O caso práctico 2 parece ser o que mellor combina o uso en tempo real coa detección e medición dos peixes. Habería que ter en conta que a inclusión de novas especies faría necesario tamén a introdución de novos patróns desa especie que permitan realizar unha correcta clasificación do obxecto detectado polo láser como peixe.

### **8.3. English**

In this thesis, three techniques have been generated that manage to validate the hypothesis of detecting the fish in the water without intrusion or physical contact with the fish through remote sensing. Using a common methodology, with computer vision techniques and several algorithms, the information acquired in each of the practical case is treated in order to reach a confirmation or rejection of that fish detection is correct, and then estimate its size.

In the first practical case, two cameras are used to generate an image with 3-dimensional data which contains information about the object and its distance from the cameras. To validate if the detected object is a fish, the fish's eye is searched in the image using Hough detection. In case of the algorithm is not able to find it, neural network will be used to find the eye.

In the second practical case, the use of a camera is complemented with an infrared sensor. By means of the infrared sensor, the object that interferes with the emitted beam is detected (sensor gets the distance between that object and itself) and at that moment an image of the object is taken, which is analyzed using computer vision to decide if it is a fish or not.

Finally, a structured light sensor that generates a point cloud with information from the scanned scenario is used. This information is combined with different types of frames provided by the device used and computer vision techniques are applied to detect and measure the fish.

Each case has been tested first in the laboratory and later in a real environment, although slightly controlled. In the future, it would be interesting to go a step further in research and be able to carry out tests in situations even closer to reality in order to adjust and refine the technique.

One of the problems in the essays and which should be keep in mind in future improvements, is being able to have more specific data on the fish that are monitored. This knowledge would be very useful to fine-tune the technique as much as possible in its measurement section. However, the closer test environment is to reality, the more difficult it will be to have this information. In other cases, this information was available but it was difficult to correlate the specimens in the images due to the number of specimens available.

After testing and analyzing the results in the practical cases, the following conclusions are generated:

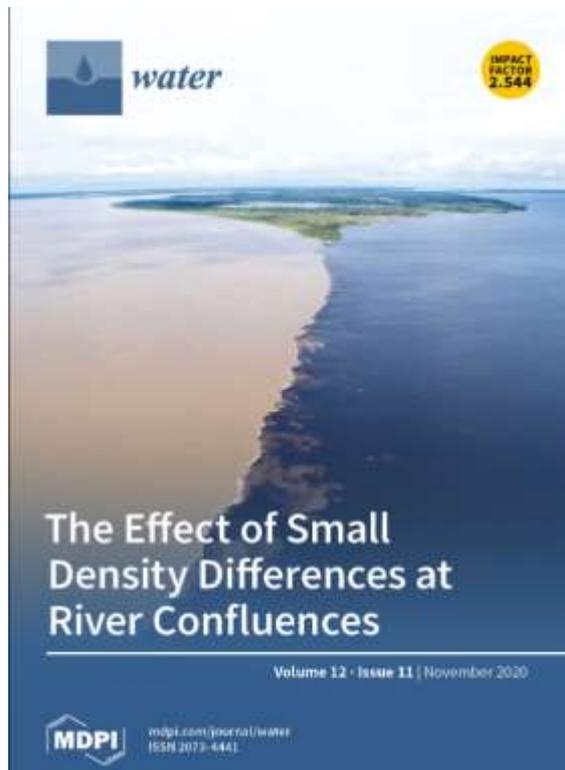
- Study case 1 with stereo vision cameras and the use of neural networks slows down their use in real time more than in study case 2. It should also be taken into account that changes in the conditions on which trained the network or the inclusion of new species would entail the retraining of the neural network.
- Practical case 3 with the Kinect has three limitations, one regarding the range of vision, another in that the device has to be placed 65 cm from the glass and finally, the fish cannot pass more than 40 cm from the glass.
- Use case 2 seems to be the one that best combines real-time use with fish detection and measurement. It should be taken into account that the inclusion of new species would also make it necessary to introduce new patterns of that species that would allow a correct classification as a fish of the object detected by the laser

## 9. Publicaciones

Durante la realización de esta tesis se han realizado diferentes publicaciones, siendo las más relevantes las que a continuación se citan:

### Revistas internacionales con impacto

Rico-Díaz, Á.J.; Rabuñal, J.R.; Gestal, M.; Mures, O.A.; Puertas, J. An Application of Fish Detection Based on Eye Search with Artificial Vision and Artificial Neural Networks. *Water* 2020, 12, 3013. Factor de impacto: 2.544; Ranking categoría JCR: 31/94 (Q2) en “Water Resources”



## Revistas internacionales con revisión por pares y sin impacto

A detection system for vertical slot fishways using laser technology and computer vision techniques (Angel J. Rico-Diaz, Alvaro Rodriguez, Daniel Villares, Juan R. Rabuñal, Jeronimo Puertas, Luis Pena) en Lecture Notes in Computer Science (ISBN 978-3-319-19257-4) Volumen 9094, págs 218-226 (2015)



### A Detection System for Vertical Slot Fishways Using Laser Technology and Computer Vision Techniques

Angel J. Rico-Diaz<sup>1,2</sup>, Alvaro Rodriguez<sup>1</sup>, Daniel Villares<sup>2</sup>,  
Juan R. Rabuñal<sup>2</sup>, Jeronimo Puertas<sup>2</sup>, and Luis Pena<sup>3</sup>

<sup>1</sup> Dept. of Information and Communication Technologies,  
University of A Coruña, Campus de Elvina, 15071 A Coruña, Spain  
(angel.j.rico, ar.rodriguez)

<sup>2</sup> Villares, Daniel, jpuertas, lpena@udc.es  
<sup>3</sup> Innovations in Construction and Civil Engineering (ICCECO),  
University of A Coruña, Campus de Elvina, 15071 A Coruña, Spain

<sup>4</sup> Dept. of Hydraulic Engineering, ETSECCP,  
University of A Coruña, Campus de Elvina, 15071 A Coruña, Spain

**Abstract.** Vertical slot fishways are structures that are placed in rivers to allow fish to avoid obstacles such as dams, hydroelectric plants. Knowing the frequency with which fish go through this type of structures can help to determine their efficiency, as well as know migratory fishes from species, determine if the fluvial course is healthy or if it is possible to fish with fauna preservation guarantee.

A non-invasive method for fish detection, without the need of direct observation, which uses a laser sensor and computer vision techniques to detect fish, is proposed in this work.

**Keywords:** Computer vision, Fishways, Fish counter, Laser detection

#### 1 Introduction

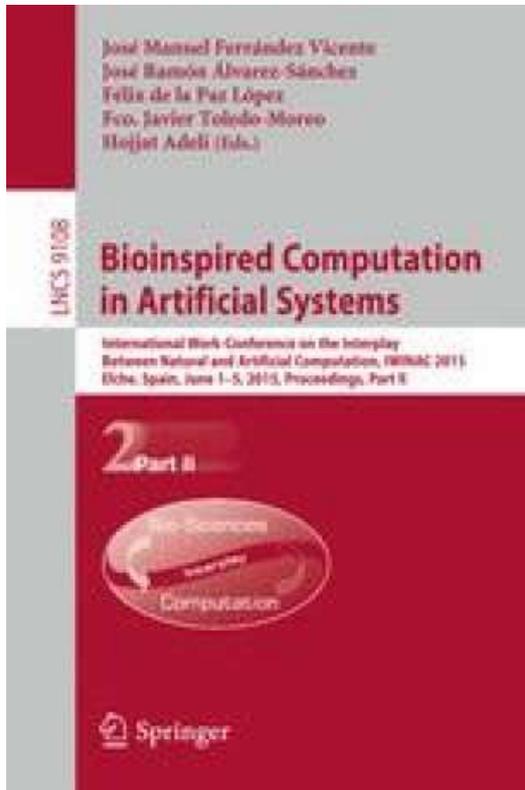
Vertical slot fishways are hydraulic structures that allow fish movement upwards in rivers with obstacles such as dams, hydroelectric plants and others.

Along the years, several researches have been conducted to study fishways and fish, including water flow features [1–3], fish swimming abilities [4, 5] or fish behavior within them [6]. In this work it is intended to determine in an easy way, in real time and with minimal cost the specimen number that go through the fishway, since knowing this value will serve to estimate the structure effectiveness [7], determine the fluvial course health, or river fish use possibilities.

One of the techniques that is used for fish detection is the sonar. The sonar is a device that detects submerged objects presence and situation through acoustic waves. From the sixties this technique has been used in applications to detect fish [8] or study fish features [9]. More recently, systems as the DIDSON (Dual-frequency identification sonar), have reduced the previous acoustic systems limitations, allowing the

© Springer International Publishing Switzerland 2015.  
J. Rojas et al. (Eds.): IWANN 2015, Part I, LNCS 9094, pp. 218–226, 2015.  
DOI: 10.1007/978-3-319-19257-4\_13

Fish monitoring and sizing using computer vision (Alvaro Rodriguez, Angel J. Rico-Diaz, Juan R. Rabuñal, Jeronimo Puertas, Luis Pena) en Lecture Notes in Computer Science (ISBN 978-3-319-18832-4) Volumen 9108, págs. 419-428 (2015)



**Fish Monitoring and Sizing Using Computer Vision**

Alvaro Rodríguez<sup>1</sup>\*, Angel J. Rico-Díaz<sup>1,2</sup>,  
Juan R. Rabuñal<sup>1,2</sup>, Jeronimo Puertas<sup>3</sup>, and Luis Pena<sup>3</sup>

<sup>1</sup> Department of Informatics and Communications Technologies,  
<sup>2</sup> Centre of Technological Innovation in Construction and Civil Engineering (CTICEC),  
<sup>3</sup> Department of Hydraulic Engineering (ETRECCP),  
University of A Coruña, Campus Elvina s/n 15071, A Coruña, Spain  
(arodriguez@unex.es, arico@unex.es, jpuertas@unex.es)

**Abstract.** This paper presents an image processing algorithm, based in a non linear 3D optical stereo system and the use of computer vision techniques, to study fish in fish tanks or ponds.  
The proposed technique will allow to study biological variables of different fish species in underwater environments.  
This knowledge may be used to replace traditional techniques such as direct observation, which are impractical or affect the fish behavior, in tasks such as aquaculture and fish farm management or fishery evaluation.  
The accuracy and performance of the proposed technique has been tested, conducting different assays with living fishes, where promising results were obtained.

**Keywords:** Segmentation · Computer vision

**1 Introduction**

Inspection of the appearance or behavior of fishes can provide with a variety of information regarding the fish species, including their health and development, and its relation with the ecosystem. In particular it can be used to provide early indications of health problems, estimating growth rates and predicting the optimal development stage to an eventual commercial exploitation.

Non-invasive fish inspection is an important question in fields related with fish studies, such as marine biology, oceanography and others. Where it is common to maintain specimens of different fish species in closed and controlled ecosystems, which need to be managed.

Additionally, fish size inspection is a critical question in fish farming applications consisting in raising fish in tanks or enclosures, usually to be used in the food industry. In this application, the optimum fish size and the parameters concerning fish growth have to be studied and monitored [1].

In these scenarios, fish are usually contained in more or less artificial structures, and the usual process to obtain the required information about fish is direct observation.

© Springer International Publishing Switzerland 2015  
J.M. Fernández-Vicente et al. (Eds.) - IWANN 2015, Part II, LNCS 9108, pp. 419–428, 2015.  
DOI: 10.1007/978-3-319-18832-4\_44

FishPath: aplicación informática de diseño de escalas de peces de hendidura vertical (María Bermúdez, Ángel Rico, Álvaro Rodríguez, Luís Pena, Juan R. Rabuñal, Jerónimo Puertas, Luis Balairón, Ángel Lara, Enrique Aramburu, Felipe Morcillo, Miriam Castillo) Ingeniería del Agua (ISSN 1134-2196), Volumen 19, Número 3, págs.179-191 (2015)



ISSN 1134-2196

**FishPath: aplicación informática de diseño de escalas de peces de hendidura vertical**  
**FishPath: computer application for design of vertical zlot fishway**

Bermúdez, M.<sup>1</sup>, Rico, A.<sup>1</sup>, Rodríguez, A.<sup>1</sup>, Pena, L.<sup>1</sup>, Rabuñal, J.R.<sup>1</sup>, Puertas, J.J., Balairón, L.J., Lara, A.<sup>1</sup>, Aramburu, E.<sup>2</sup>, Morcillo, F.<sup>3</sup> y Castro, M.<sup>4</sup>

<sup>1</sup> I.T.A. Ingeniería de Caminos, Canales y Puertos, Universidad de Cantabria, Campus de Siles, s/n 39011, S.Cantabria, España; <sup>2</sup> mbermudez@ucn.es; <sup>3</sup> jpuertas@ucn.es; <sup>4</sup> fcastro@ucn.es; <sup>5</sup> aramburu@ucn.es; <sup>6</sup> morcillo@ucn.es; <sup>7</sup> jrabuñal@ucn.es; <sup>8</sup> lbalairon@ucn.es; <sup>9</sup> alara@ucn.es; <sup>10</sup> lpena@ucn.es; <sup>11</sup> a.rico@ucn.es; <sup>12</sup> a.lara@ucn.es; <sup>13</sup> e.aramburu@ucn.es; <sup>14</sup> m.castillo@ucn.es; <sup>15</sup> m.bermudez@ucn.es

REVISADO: 2014-03-10 ACEPTADO: 2014-03-10 PUBLICADO: 2015-03-10

**RESUMEN**

Las escalas de peces de hendidura vertical son estructuras hidráulicas diseñadas para permitir la migración de los peces río arriba. A través de estructuras en forma de Z, se crean zonas de flujo que permiten superar de la manera más eficiente las barreras físicas y biológicas, dado que las estructuras hidráulicas de la escala deben cumplir a su vez con los requisitos de las especies objetivo. En este artículo se presenta una aplicación informática que ayuda a diseñar la estructura de un modelo de escala de acuerdo con criterios biológicos. El modelo simulador está basado en el método de los elementos finitos (FEM) y se basa en la ecuación de Navier-Stokes y en sus variantes en estado estacionario. Permite así comparar diseños de forma rápida y obtener un modelo preliminarmente más eficiente. Con esta herramienta se busca facilitar a los técnicos en el diseño de un modelo de escala de peces de hendidura vertical la obtención de diseños hidráulicos, que respondan a las necesidades de las investigaciones más recientes en este campo.

**Palabras clave:** Simulación de flujo; Cálculo de flujo de un pez; Diseño hidráulico.

**ABSTRACT**

Vertical zlot fishways are hydraulic structures designed to allow fish migration through channels in rivers. The appropriate design of these devices depends on an iterative process of hydraulic and biological variables, since the hydrodynamic structure of the fishway must meet the requirements of the fish species for which it is intended. The paper presents a computer application to analyze the efficiency of a fishway model according to biological criteria. The program allows rapid generation of models in the design process. In this article, a computer application is presented that helps to design the structure of a model of a vertical zlot fishway according to biological criteria. The model simulator is based on the finite element method (FEM) and is based on the Navier-Stokes equation and its variants in steady state. This allows comparing designs quickly and obtaining a more efficient preliminary model. With this tool, the aim is to facilitate to technicians in the design of a model of a vertical zlot fishway the obtaining of hydraulic designs, which respond to the needs of the most recent research in this field.

**Key words:** Flow simulation; performance hydraulic design.

0949-4931/15/1903-0179-13

DOI: 10.1016/j.ia.2014.12.004

Un sistema de detección de peces para escala de hendidura vertical utilizando tecnología láser y técnicas de visión artificial (Angel J. Rico-Díaz, Juan R. Rabuñal, Jerónimo Puertas, Luis Pena, Álvaro Rodríguez) en Ingeniería del Agua (ISSN: 1134-2196). Ingeniería del Agua (ISSN 1134-2196), Volumen 19, Número 4, págs.229-239 (2015)



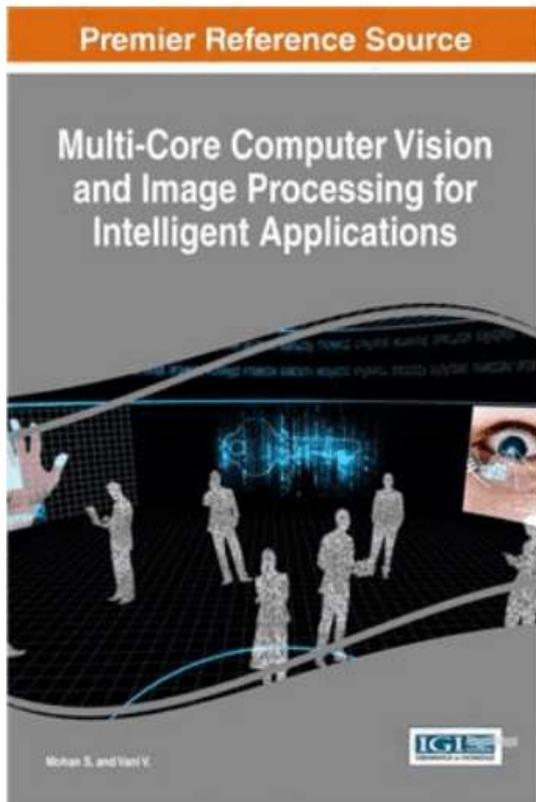
## Congresos internacionales

Comunicación oral en el 13th International Work-Conference on Artificial Neural Networks (IWANN) 2015 (10-12/06/2015 en Palma de Mallorca) con el título “A detection system for vertical slot fishways using laser technology and computer vision techniques”. Autores: Angel J. Rico-Díaz, Alvaro Rodriguez, Daniel Villares, Juan R. Rabuñal, Jeronimo Puertas, Luis Pena

Comunicación oral en el Work-Conference on the Interplay between Neural and Artificial Computation (IWINAC) 2015 (01-05/06/2015 en Elche) con el título “Fish monitoring and sizing using Computer Vision”. Autores Alvaro Rodriguez, Angel J. Rico-Díaz, Juan R. Rabuñal, Jeronimo Puertas, Luis Pena

## Capítulos de libros

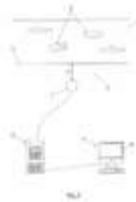
Alvaro Rodríguez, Angel Jose Rico-Díaz, Juan Ramón Rabuñal Dopico, Marcos Gestal: Fish Tracking with Computer Vision Techniques: An Application to Vertical Slot Fishways. In: Mohan, S., Vani, V. (eds.) Multi-Core Computer Vision and Image Processing for Intelligent Applications. pp. 74-104. IGI Global, Hershey, PA, USA (2017)





# Patentes

## ES 2 649 056 A1: Sistema de detección y medición de peces y objetos móviles bajo el agua

 <p>OFICINA ESPAÑOLA DE PATENTES Y MARCAS ESPAÑA</p>		 <p>① Número de publicación: <b>2 649 056</b> ② Número de solicitud: 201000535 ③ Int. Cl.: G01S 17/42 G06K 9/00</p>
④ SOLICITUD DE PATENTE		A1
⑤ Fecha de presentación: 07.07.2016	⑥ Fecha de publicación de la solicitud: 09.05.2018	⑦ Solicitante: UNIVERSIDADE DA CORUÑA (100.0%) C/O - Universidade da Coruña, Edificio de Servicios Centrales de Investigación Campus de Eliña s/n 15071 A Coruña ES ⑧ Inventores: RABUÑAL DOPICO, Juan Ramón; PUERTAS AGUDO, Jerónimo; RODRÍGUEZ TAJES, Álvaro; RICO DÍAZ, Ángel José; CONDE VÁZQUEZ, André y PALLAS FERNÁNDEZ, Adrián
⑨ Título: Sistema de detección y medición de peces y objetos móviles bajo el agua		
⑩ Resumen: Sistema de detección y medición de peces y objetos móviles bajo el agua, que comprende un escáner 3D (1) para la medición mediante emisión de luz de la distancia de los puntos de una escena (2), en un entorno acuático (7), una unidad de procesamiento de datos (5) configurada para generar información de la posición y tamaño de los objetos móviles detectados en el entorno acuático (7) a partir de las señales recibidas por el escáner 3D (1), y una unidad de control de usuario (4) para la representación en una pantalla (3) de los datos de posición y tamaño de los objetos detectados. Se trata de un sistema no invasivo para las especies donde los peces detectados son seguidos correctamente hasta dejar de ser visibles, son separados en el caso de estar solapados y son medidos correctamente.		

ES 2 649 056 A1

## Otras publicaciones

Comunicación oral en el VI Seminario sobre las Líneas Prioritarias de la Red de Laboratorios de Hidráulica de España (9 de Julio de 2013 en Madrid) con el título “Ensayos de detección de peces para escalas de hendidura vertical utilizando escáner láser 2D”. Autores: Ángel Rico, Juan R. Rabuñal, Alvaro Rodríguez, Daniel Villares, Luis Pena, Jerónimo Puertas.

Ensayos de detección de peces para escalas de hendidura vertical utilizando escáner láser 2d (Ángel Rico, Juan R. Rabuñal, Alvaro Rodríguez, Daniel Villares, Luis Pena, Jerónimo Puertas) en Actividad experimental de I+D+i en ingeniería hidráulica en España. Red de Laboratorios de hidráulica en España. Seminario 2013 (ISBN: 978-84-943649-5-2), 2015

Comunicación oral en el VII Seminario sobre las Líneas Prioritarias de la Red de Laboratorios de Hidráulica de España (10 de junio de 2014 en Madrid) con el título “Sistema de detección de peces para escalas de hendidura vertical utilizando tecnología escáner láser y visión artificial”. Autores: Ángel Rico, Juan R. Rabuñal, Alvaro Rodríguez, Daniel Villares, Luis Pena, Jerónimo Puertas.

Sistema de detección de peces para escalas de hendidura vertical utilizando tecnología escáner láser y visión artificial (Ángel Rico, Juan R. Rabuñal, Alvaro Rodríguez, Daniel Villares, Luis Pena, Jerónimo Puertas) en Actividad experimental de I+D+i en ingeniería hidráulica en España. Red de Laboratorios de hidráulica en España. Seminario 2014 (ISBN: 978-84-943649-6-9), 2015

Comunicación oral en las Jornadas Internacionales de Investigación Biomédica en Galicia, GIB (Grupo de Investigación Biomédica en Galicia) 25 años de Historia (25 y 26 de mayo de 2015 en A Coruña) con el título “Visión artificial con sensores láser y visión estereoscópica”. Autor: Ángel J. Rico-Díaz

Comunicación oral en el VIII Seminario sobre las Líneas Prioritarias de la Red de Laboratorios de Hidráulica de España (25 de junio de 2015 en A Coruña) con el título

“Comparativa de diferentes técnicas ópticas para la detección de peces”. Autores: Ángel Rico, Juan R. Rabuñal, Alvaro Rodríguez, Luis Pena, Jerónimo Puertas.

Comunicación oral en el VIII Seminario sobre las Líneas Prioritarias de la Red de Laboratorios de Hidráulica de España (25 de junio de 2015 en A Coruña) con el título “Comparativa de diferentes técnicas ópticas para la detección de peces”. Autores: Ángel Rico, Juan R. Rabuñal, Alvaro Rodríguez, Luis Pena, Jerónimo Puertas.

Comunicación oral en las Jornadas Internacionales de Investigación Biomédica en Galicia, GIB (Grupo de Investigación Biomédica en Galicia) 25 años de Historia (25 y 26 de mayo de 2015 en A Coruña) con el título “Visión artificial con sensores láser y visión estereoscópica”. Autor: Ángel J. Rico-Díaz

Comunicación en las V Jornadas de Ingeniería del Agua JIA 2017 con el título “Sistema de detección y medición de peces en nubes de puntos para escalas de peces de hendidura vertical”. Autores: Rico-Díaz, A.J, Pallas A., Rabuñal J.R., Puertas, J.

# Bibliografía

- [1] T. Castro-Santos, A. Haro, and S. Walk, "A passive integrated transponder (PIT) tag system for monitoring fishways," *Fisheries Research*, vol. 28, pp. 253-261, 1996.
- [2] M. Ashraf and J. Lucas, "Underwater Object Recognition Techniques Using Ultrasonics," presented at the IEEE Int. Conf. Oceans 94 Osates, Brest, 1994.
- [3] S. A. A. Bakar, N. R. Ong, M. H. A. Aziz, J. B. Alcain, W. M. W. N. Haimi, and Z. Sauli, "Underwater detection by using ultrasonic sensor," *AIP Conference Proceedings*, vol. 1885, p. 020305, 2017/09/26 2017.
- [4] M. Ghobrial, "Fish Detection Automation from ARIS and DIDSON SONAR Data," University of Oulu, 2019.
- [5] R. Schettini and S. Corchs, "Underwater Image Processing: State of the Art of Restoration and Image Enhancement Methods," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, p. 746052, 2010/04/21 2010.
- [6] E. Trucco and A. T. Olmos-Antillon, "Self-Tuning Underwater Image Restoration," *IEEE Journal of Oceanic Engineering*, vol. 31, pp. 511-519, 2006.
- [7] A. Yamashita, M. Fujii, and T. Kaneko, "Color Registration of Underwater Images for Underwater Sensing with Consideration of Light Attenuation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 4570-4575.
- [8] D. Lee, G. Kim, D. Kim, H. Myung, and H.-T. Choi, "Vision-based object detection and tracking for autonomous navigation of underwater robots," *Ocean Engineering*, vol. 48, pp. 59-68, 2012/07/01/ 2012.
- [9] J. Puertas, L. Pena, and T. Teijeiro, "Experimental Approach to the Hydraulics of Vertical Slot Fishways," *Journal of Hydraulics Engineering*, vol. 130 (1), pp. 10-23, 2004.
- [10] L. Tarrade, A. Texier, and L. David, "Topologies and measurements of turbulent flow in vertical slot fishways," *Hydrobiologia*, vol. 609, pp. 177-188, 2008.

- [11] S. Wu, N. Rajaratnam, and C. Katopodis, "Structure of flow in vertical slot fishways," *Journal of Hydraulic Engineering*, vol. 125 (4), pp. 351-360, 1999.
- [12] R. W. Blake, "Fish functional design and swimming performance," *Journal of Fish Biology*, vol. 65, pp. 1193-1222, 2004.
- [13] H. Dewar and J. Graham, "Studies of tropical tuna swimming performance in a large water tunnel - Energetics," *Journal of Experimental Biology*, vol. 192, pp. 13-31, 1994.
- [14] A. Rodríguez, M. Bermúdez, J. Rabuñal, Aff.M.ASCE, J. Puertas, J. Dorado, *et al.*, "Optical Fish Trajectory Measurement in Fishways through Computer Vision and Artificial Neural Networks," *Journal of Computing in Civil Engineering*, vol. 25, pp. 291-301, 2011.
- [15] F. Bernardini and H. Rushmeier, "The 3D Model Acquisition Pipeline," *Comput. Graph. Forum*, vol. 21, pp. 149-172, 06/01 2002.
- [16] L. Li, "Time-of-Flight Camera - An Introduction," *Texas Instruments - Technical White Paper*, 2014.
- [17] D. Fofi, T. Sliwa, and Y. Voisin, "A comparative survey on invisible structured light," in *Proc.SPIE*, 2004.
- [18] X. Ning, X. Zhang, Y. Wang, and M. Jaeger, "Segmentation of architecture shape information from 3D point cloud," presented at the Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry, Yokohama, Japan, 2009.
- [19] R. Lu, I. Brilakis, and C. R. Middleton, "Detection of Structural Components in Point Clouds of Existing RC Bridges," *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, pp. 191-212, 2019/03/01 2019.
- [20] Q. Wang and M.-K. Kim, "Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018," *Advanced Engineering Informatics*, vol. 39, pp. 306-319, 2019/01/01/ 2019.
- [21] J. Xu, N. Xi, C. Zhang, Q. Shi, and J. Gregory, "Real-time 3D shape inspection system of automotive parts based on structured light pattern," *Optics & Laser Technology*, vol. 43, pp. 1-8, 2011/02/01/ 2011.
- [22] C. Zhang, S. Pujades, M. J. Black, and G. Pons-Moll, "Detailed, accurate, human shape estimation from clothed 3D scan sequences," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4191-4200.

- [23] B. Gorte and D. Winterhalder, "Reconstruction of laser-scanned trees using filter operations in the 3D raster domain," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, p. W2, 2004.
- [24] M. M. Saberioon and P. Cisar, "Automated multiple fish tracking in three-Dimension using a Structured Light Sensor," *Computers and Electronics in Agriculture*, vol. 121, pp. 215-221, 2016/02/01/ 2016.
- [25] J.-J. Hernández-López, A.-L. Quintanilla-Olvera, J.-L. López-Ramírez, F.-J. Rangel-Butanda, M.-A. Ibarra-Manzano, and D.-L. Almanza-Ojeda, "Detecting objects using color and depth segmentation with Kinect sensor," *Procedia Technology*, vol. 3, pp. 196-204, 2012/01/01/ 2012.
- [26] L. Xia, C. Chen, and J. K. Aggarwal, "Human detection using depth information by Kinect," in *CVPR 2011 WORKSHOPS*, 2011, pp. 15-22.
- [27] L. Yang, Y. Liu, H. Yu, X. Fang, L. Song, D. Li, *et al.*, "Computer Vision Models in Intelligent Aquaculture with Emphasis on Fish Detection and Behavior Analysis: A Review," *Archives of Computational Methods in Engineering*, vol. 28, pp. 2785-2816, 2021/06/01 2021.
- [28] A. Rodríguez, "A Methodology to Develop Computer Vision Systems in Civil Engineering: Applications in Material Testing and Fish Tracking," Universidade da Coruña. Departamento de Tecnoloxías da Información e as Comunicacóns, 2014.
- [29] V. Shevchenko, T. Eerola, and A. Kaarna, *Fish Detection from Low Visibility Underwater Videos*, 2018.
- [30] E. Harveya, M. Cappob, M. Shortisc, S. Robsond, J. Buchanane, and P. Speareb, "The accuracy and precision of underwater measurements of length and maximum body depth of southern bluefin tuna (*Thunnus maccoyii*) with a stereo-video camera system," *Fisheries Research*, vol. 63, pp. 315-326, 2003.
- [31] M. Ravanbakhsh, M. Shortis, F. Shafait, A. Mian, E. Harvey, and J. Seager, "Automated Fish Detection in Underwater Images Using Shape-Based Level Sets," *The Photogrammetric Record*, vol. 30, pp. 46-62, 2015.
- [32] M. R. Shortis, M. Ravanbakhsh, F. Shaifat, E. S. Harvey, A. Mian, J. W. Seager, *et al.*, "A review of techniques for the identification and measurement of fish in underwater stereo-video image sequences," 2013, pp. 87910G-87910G-10.

- [33] H.-J. Sung, M.-K. Park, and J. W. Choi, "Automatic Grader for Flatfishes Using Machine Vision," *International Journal of Control, Automation and Systems*, vol. 18, pp. 3073-3082, 2020/12/01 2020.
- [34] R. E. Craig and S. T. Forbes, "Design of a sonar for fish counting," *Fiskeridirektoratets skrifter, Serie Havundersøkelser*, vol. Vol 15 no 3, pp. 210-219, 1969.
- [35] J. E. Ehrenberg, "A method for extracting the fish target strength distribution from acoustic echoes," presented at the Engineering in the Ocean Environment, Ocean 72 Newport, RI, USA, 1972.
- [36] H. Balk and T. Lindem, "Improved fish detection in data from split-beam sonar," *Aquatic Living Resources*, vol. 13, pp. 297-303, 2000.
- [37] E. Belcher, M. B., and T. G, "Object identification with acoustic lenses," presented at the OCEANS, 2001. MTS/IEEE Conference and Exhibition, Honolulu, HI, 2001.
- [38] J. Han, N. Honda, A. Asada, and K. Shibata, "Automated acoustic method for counting and sizing farmed fish during transfer using DIDSON," *Fisheries Science*, vol. 75, pp. 1359-1367, 2009.
- [39] J. A. Holmes, G. M. W. Cronkite, H. J. Enzenhofer, and T. J. Mulligan, "Accuracy and precision of fish-count data from a "dual-frequency identification sonar" (DIDSON) imaging system," *ICES Journal of Marine Science*, vol. 63, pp. 543-555, 2006.
- [40] V. Mitra, C.-J. Wang, and S. Banerjee, "Lidar Detection of Underwater Objects Using a Neuro-SVM-Based Architecture," *IEEE TRANSACTIONS ON NEURAL NETWORKS*, 2004.
- [41] L. Baumgartner, M. Bettanin, J. McPherson, M. Jones, B. Zampatti, and K. Beyer, "Assessment of an infrared fish counter (Vaki Riverwatcher) to quantify fish migrations in the Murray-Darling Basin," *Industry & Investment NSW, Fisheries Final Report Series* vol. 116, p. 47, 2010.
- [42] (08/01/2021). *Vaki*. Available: <https://vakiiceland.is/>
- [43] (01/06/2021). *AquaScan*. Available: <https://www.aquascan.com/>
- [44] (01/06/2021). *Calitri technology*. Available: <https://www.calitri-technology.com/en/home-2/>
- [45] (01/06/2021). *Faivre*. Available: <http://www.faivre.fr/index.php/en/>
- [46] (01/06/2021). *Impex Agency ApS* Available: <http://impexagency.dk/>

- [47] (01/06/2021). *Smith-Root*. Available: <https://www.smith-root.com/>
- [48] (2020-11-22). *Sherlock*. Available: <https://www.teledynedalsa.com/en/products/imaging/vision-software/sherlock/>
- [49] (2020-11-22). *Halcon*. Available: <https://www.mvtec.com/products/halcon/>
- [50] (2020-11-22). *Point Cloud Library*. Available: <https://pointclouds.org/>
- [51] *OPENCV: Open Source Computer Vision*. Available: <http://opencv.org>
- [52] *.NET Framework*. Available: [https://msdn.microsoft.com/library/hh425099\(v=vs.110\).aspx](https://msdn.microsoft.com/library/hh425099(v=vs.110).aspx)
- [53] *EmguCV*. Available: [http://www.emgu.com/wiki/index.php/Main\\_Page](http://www.emgu.com/wiki/index.php/Main_Page)
- [54] M. Moniruzzaman, S. M. S. Islam, M. Bennamoun, and P. Lavery, "Deep Learning on Underwater Marine Object Detection: A Survey," in *Advanced Concepts for Intelligent Vision Systems*, Cham, 2017, pp. 150-160.
- [55] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779-788.
- [56] M. Sung, S. Yu, and Y. Girdhar, "Vision based real-time fish detection using convolutional neural network," in *OCEANS 2017 - Aberdeen*, 2017, pp. 1-6.
- [57] S. Cui, Y. Zhou, Y. Wang, and L. Zhai, "Fish Detection Using Deep Learning," *Applied Computational Intelligence and Soft Computing*, vol. 2020, pp. 1-13, 01/23 2020.
- [58] A. F. A. Fernandes, E. M. Turra, É. R. de Alvarenga, T. L. Passafaro, F. B. Lopes, G. F. O. Alves, *et al.*, "Deep Learning image segmentation for extraction of fish body measurements and prediction of body weight and carcass traits in Nile tilapia," *Computers and Electronics in Agriculture*, vol. 170, p. 105274, 2020/03/01/ 2020.
- [59] F. Suo, K. Huang, G. Ling, Y. Li, and J. Xiang, "Fish Keypoints Detection for Ecology Monitoring Based on Underwater Visual Intelligence," in *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2020, pp. 542-547.
- [60] L. Yanjun, Huang Kangwei, and X. Ji, "Measurement of dynamic fish dimension based on stereoscopic vision," *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)* vol. 36, pp. 220-226, 2020.
- [61] C. Costa, M. Scardi, V. Vitalini, and S. Cataudella, "A dual camera system for counting and sizing Northern Bluefin Tuna (*Thunnus thynnus*; Linnaeus, 1758) stock,

during transfer to aquaculture cages, with a semi automatic Artificial Neural Network tool," *Aquaculture*, vol. 291, pp. 161-167, 2009.

[62] A. Salman, S. A. Siddiqui, F. Shafait, A. Mian, M. R. Shortis, K. Khurshid, *et al.*, "Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system," *ICES Journal of Marine Science*, vol. 77, pp. 1295-1307, 2020.

[63] A. Jalal, A. Salman, A. Mian, M. Shortis, and F. Shafait, "Fish detection and species classification in underwater environments using deep learning with temporal information," *Ecological Informatics*, vol. 57, p. 101088, 2020/05/01/ 2020.

[64] L. Liu, H. Lu, Z. Cao, and Y. Xiao, "Counting Fish in Sonar Images," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3189-3193.

[65] P. Muñoz-Benavent, V. Puig-Pons, G. Andreu-García, V. Espinosa, V. Atienza-Vanacloig, and I. Pérez-Arjona, "Automatic Bluefin Tuna Sizing with a Combined Acoustic and Optical Sensor," *Sensors*, vol. 20, 2020.

[66] X. Yang, S. Zhang, J. Liu, Q. Gao, S. Dong, and C. Zhou, "Deep learning for smart fish farming: applications, opportunities and challenges," *Reviews in Aquaculture*, vol. 13, pp. 66-90, 2021/01/01 2021.

[67] R. I. Hartley and A. Zisserman, Eds., *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004, p.^pp. Pages.

[68] F. H. Abad, V. H. Abad, J. F. Andreu, and M. O. Vives, "Application of Projective Geometry to Synthetic Cameras.," presented at the XIV International Conference of Graphic Engineering, 2002.

[69] N. Martin, B. A. Perez, D. G. Aguilera, and J. G. Lahoz, "Applied Analysis of Camera Calibration Methods for Photometric Uses.," presented at the VII National Conference of Topography and Cartography, 2004.

[70] Z. Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations," presented at the International Conference on Computer Vision (ICCV), 1999.

[71] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, Greece, 1999, pp. 1197-1203.

- [72] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol. 24, pp. 603-619, 2002.
- [73] S. Suzuki and K. Abe, "Topological Structural Analysis of Digitized Binary Images by Border Following," *Computer Vision, Graphics, and Image Processing*, vol. 30, pp. 32-46, 1985.
- [74] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 679-698, 1986.
- [75] D. Nistér and H. Stewénius, "Linear Time Maximally Stable Extremal Regions," in *Computer Vision – ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part II*, D. Forsyth, P. Torr, and A. Zisserman, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 183-196.
- [76] J. Illingworth and J. Kittler, "The adaptive hough transform," *IEEE Trans Pattern Anal Mach Intell*, vol. 9, p. 8, 1987.
- [77] H. K. Yuen, J. Princen, J. Illingworth, and J. Kittler, "Comparative study of Hough Transform methods for circle finding," *Image and Vision Computing*, vol. 8, pp. 71-77, 1990/02/01 1990.
- [78] T. D. Lamb, S. P. Collin, and E. N. Pugh, "Evolution of the vertebrate eye: opsins, photoreceptors, retina and eye cup," *Nat Rev Neurosci*, vol. 8, pp. 960-976, 12//print 2007.
- [79] M. F. Land and D.-E. Nilsson, *Animal Eyes*. Oxford: Oxford University Press, 2012.
- [80] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359-366, 1989/01/01/ 1989.
- [81] (2020-11-11). *Hokuyo Laser*. Available: [https://www.hokuyo-aut.jp/02sensor/07scanner/urg\\_04lx\\_ug01.html](https://www.hokuyo-aut.jp/02sensor/07scanner/urg_04lx_ug01.html)
- [82] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image Analysis Using Mathematical Morphology," *IEEE TRANSACTIONS ON PATTERN AND MACHINE INTELLIGENCE*, vol. PAMI-9, No 4, pp. 532-550, 1987.
- [83] R. R. Morales and J. H. S. Azuela, *Procesamiento y análisis digital de imágenes*: Ra-Ma, 2011.

- [84] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *Systems, Man and Cybernetics, IEEE Transactions*, vol. 9, pp. 62-66, 1979.
- [85] G. Pajares Martinsanz and J. De la Cruz García, *Visión por computador imágenes digitales y aplicaciones*, 2<sup>a</sup> ed.: Ra-Ma, 2007.
- [86] J. Ochoa Somuanom, C. Pérez Lara, J. H. Toscano Martínez, and C. G. Pereyra Ramos, "Clasificación de objetos rígidos a partir de imágenes digitales empleando los momentos invariantes de Hu," presented at the X Congreso Internacional sobre Innovación y Desarrollo Tecnológico, Cuernavaca Morelos, México, 2013.
- [87] A. Pallas Fernandez, "Diseño e implementación de un sistema de detección y medición de peces en nubes de puntos," ed: Universidade da Coruña, 2015.
- [88] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 2004, pp. 28-31 Vol.2.
- [89] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, 1998/06/01 1998.