ΕΛΛΗΝΙΚΟ ΜΕΣΟΓΕΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΡΟΗΓΜΕΝΑ ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΓΩΓΗΣ, ΑΥΤΟΜΑΤΙΣΜΟΥ & ΡΟΜΠΟΤΙΚΗΣ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Εκτίμηση Μεγέθους και Σχήματος Ψαριών Μέσω Στερεοσκοπικής Όρασης

Βοσκάκης Δημήτριος

Εσωτ. Επιβλέποντες: Δρ. Μακρής Αλέξανδρος
Δρ. Σφακιωτάκης Μιχαήλ

Εξωτ. Επιβλέπων: Δρ. Παπανδρουλάκης Νικόλαος
Διευθυντής Ερευνών,
Ελληνικό Κέντρο Θαλάσσιων Ερευνών

Ηράκλειο Κρήτης
Ιούνιος 2019

HELLENIC MEDITERRANEAN UNIVERSITY
SCHOOL OF ENGINEERING

POSTGRADUATE PROGRAM
MASTER OF SCIENCE IN
ADVANCED MANUFACTURING SYSTEMS, AUTOMATION AND ROBOTICS

MASTER THESIS

Fish Size and Shape Estimation with Stereoscopic Vision

Voskakis Dimitrios

Internal Supervisors: Dr. Makris Alexandros
Dr. Sfakiotakis Michael

External Supervisor: Dr. Papandroulakis Nikolaos
Research Director,
Hellenic Center for Marine Research

Heraklion Crete
June 2019

Acknowledgement

The present thesis was carried out under the collaboration of the Hellenic Mediterranean University (former Technological Educational Institute of Crete) and the Hellenic Center for Marine Research, in Heraklion, Crete. It was contacted at the Institute of Marine Biology Biotechnology and Aquaculture of the Hellenic Center for Marine Research (HCMR), partly financed from the project AQUAEXCEL (H2020-INFRAIA-1-2014-2015, AQUAculture infrastructures for EXCELlence in European fish research towards 2020 (652831)) and from the project PerformFISH (H2020-SFS-23-2016 "Consumer driven Production: Integrating Innovative Approaches for Competitive and Sustainable Performance across the Mediterranean Aquaculture Value Chain" PerformFISH (727610)). The subject of the thesis was fish size and shape estimation with stereoscopic vision.

The study was implemented under the supervision of Dr. Sfakiotakis Michalis, Associate Professor at the Electrical and Computer Engineering Department of the Hellenic Mediterranean University (former Technological Educational Institute of Crete), Dr. Papandroulakis Nikos, Research Director at Hellenic Center for Marine Research (HCMR), Dr. Alexandros Makris, Affiliated Researcher at the Institute of Computer Science of the Foundation for Research and Technology-Hellas (ICS-FORTH), all of who I would like to thank for their support and trust.

I am also grateful to all the staff at IMBBC-HCMR, and particularly Mr. Chalkiadakis Vaggelis for their help during my study and to my family for their support all this time.

Abstract

Aquaculture industry is the faster growing sector of primary production the recent years, having a significant role in the global food production. To improve its sustainability, the industries invest in technologies and innovative systems towards methodologies of high precision. One of the most crucial issue in an aquaculture farm is the accurate, continues estimation of fish biomass during on growing as this defined several husbandry parameters that affects the final production and the operational cost.

In the current project, a new length estimation system through stereoscopy has been studied. Paired cameras mounted in a waterproof housing receive a sequence of images. The covering is made of special protective material, safeguarding the system from high pressures caused by water environment. A minicomputer operates the cameras, receiving the necessary data and transmitting it to the main system for further analysis.

Based on the stereoscopic method, where two overlapping images, that when slightly separated, can give the third dimension, the depth, the main system measures the necessary information for the measurement of the fish. Staring from the image captured a candidate fish is recognized and spotted. Following this, through various stages of image processing, we can estimate the size of the candidate fish. A three dimensional fish model is used both for the recognition and the measurement of the fish. We use a fish model which has a skeleton consisting of joints and links, adjusting it accordingly on the candidate fish, taking its body orientations into consideration at the same time. The visualization of this three dimensional model is based on artificial intelligence and more specifically on genetic algorithms. This algorithm, called Particle Swarm Optimization, optimizes a problem through a repeated improvement of a solution.

The accuracy of the system was validated with a field trial involving live fish inside a tank, and the results indicated the efficacy and potential of the proposed method.

Περίληψη

Η υδατοκαλλιέργεια αποτελεί τα τελευταία χρόνια έναν από τους πιο ταχέως αναπτυσσόμενους παραγωγικούς κλάδους, κατέχοντας έναν πολύ σημαντικό ρόλο στην παγκόσμια παραγωγή τροφίμων. Για να βελτιώσουν την βιωσιμότητα τους, οι βιομηχανίες του χώρου επενδύουν σε τεχνολογίες και καινοτόμα συστήματα με μεθόδους υψηλής ακρίβειας. Ένα από τα πιο κρίσιμα ζητήματα σε ένα αγρόκτημα υδατοκαλλιέργειας είναι η ακριβής και συνεχής εκτίμηση της βιομάζας των ψαριών κατά τη διάρκεια της καλλιέργειας, καθώς αυτό ορίζει αρκετές παραμέτρους κτηνοτροφίας που επηρεάζουν την τελική παραγωγή και το λειτουργικό κόστος.

Στην παρούσα εργασία, μελετάται μια νέα μέθοδος εκτίμησης του μήκους των ψαριών μέσω στερεοσκοπικής μεθόδου, όπου ένα ζεύγος καμερών λαμβάνει ζεύγη εικόνων. Οι κάμερες είναι τοποθετημένες σε στεγανό περίβλημα, φτιαγμένο από ειδικό υλικό, που προστατεύει το σύστημα από τις υψηλές πιέσεις που ασκούνται από το νερό. Ένας μικρο-υπλογιστής διαχειρίζεται τις κάμερες, λαμβάνοντας τα απαραίτητα δεδομένα και αποστέλλοντας τα στο κύριο σύστημα για περαιτέρω ανάλυση.

Με βάση τη στερεοσκοπική μέθοδο, όπου δύο επικαλυπτόμενες εικόνες, που όταν είναι ελαφρώς διαχωρισμένες μπορούν να δώσουν την τρίτη διάσταση, το βάθος, το κύριο σύστημα υπολογίζει τις απαραίτητες πληροφορίες για τη μέτρηση των ψαριών. Εξετάζοντας τις εικόνες που λαμβάνονται από τις κάμερες, το σύστημα εντοπίζει και αναγνωρίζει σε αυτές το υποψήφιο προς μέτρηση ψάρι. Κατόπιν, και μετά από την εφαρμογή διαφόρων σταδίων επεξεργασίας της εικόνας, μπορούμε να υπολογίσουμε το μέγεθος του υποψήφιου ψαριού. Αυτό επιτυγχάνεται με τη χρήση ενός τρισδιάστατου πολυ-αρθρωτού μοντέλου ψαριού. Το μοντέλο είναι σε κλίμακα, όπου ο σκελετός του μπορεί να αυξομειώνει το μήκος του ταιριάζοντας το κατάλληλα επάνω στο υποψήφιο ψάρι, εξετάζοντας και τυχόν μεταβολές στις γωνίες των αρθρώσεων μεταξύ των συνδέσμων που απαρτίζουν τον σκελετό. Η βελτιστοποίηση του τρισδιάστατου μοντέλου βασίζεται στη τεχνητή νοημοσύνη και συγκεκριμένα στον αλγόριθμο Particle Swarm Optimization, ο οποίος βελτιστοποιεί ένα πρόβλημα μέσω επανειλημμένων προσπαθειών βελτίωσης μιας υποψήφιας λύσης.

Η ακρίβεια του συστήματος αποτιμήθηκε μέσω σειράς δοκιμών, συμπεριλαμβανομένων και πειραμάτων με ψάρια εντός μίας δεξαμενής, τα αποτελέσματα των οποίων είναι ιδιαίτερα ενθαρρυντικά για την ακρίβεια και αποτελεσματικότητα της προτεινόμενης μεθόδου.

Contents

Acronyms

PSO: Particle Swarm Optimization
GA: Genetic Algorithm
FAO: Food and Agriculture Organization
GLCM: Gray-Level Co-Occurrence Matrix
POE: Power Over Ethernet
AI: Artificial Intelligence
3D: 3 Dimensional
TL: Total Length
FL: Fork Length
CPU: Central Processing Unit
GB: Gigabyte
RAM: Random Access Memory
RGB: Red Green Blue
C: Optical Center
P: Image in 3 dimensional coordinates
p: Image in 2 dimensional coordinates
f: Focal Length
t: Transformation Matrix (3x1)
R: Orientation Matrix (3x3)
B: Baseline (Distance between two cameras)
Z: Depth Observation
A: Camera Intrinsic Matrix
fps: frames per second

Chapter 1: Introduction

1.1 Aquaculture

Aquaculture refers to animal production in controlled conditions both in freshwater and marine environment. Aquaculture is the most fast developing sector of primary production world-wide. As shown in Fig 1, it is estimated by FAO that farmed fish production will increase from 45% to 62% between 2009 and 2030. Demand for aquatic products continually increases globally and according to FAO estimates, the needs gap will reach 30 millions of tons by 2030. The only profitable and environmentally friendly option is the further development of aquaculture. Something which is particularly true for the finfish aquaculture

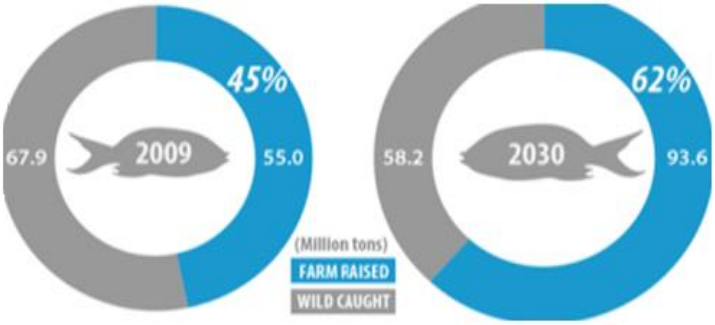


Fig 1.Predictions chart for aquaculture

Finfish aquaculture is a well-developed industry in Greece contributing to national exports with aps 500M€ annually (FGM, 2018) competing with the olive oil for the highest export value product. The Greek sector, taking advantage of the geographic specificities of the country, can with proper management, further develop with increased production, thus contributing to the general food demand and also to the national economy.

Aquaculture has been a major activity in the Greek economy for the last 30 years. Based on reports by the Federation of Greek Maricultures the main species reared are the European sea bass (*Dicentrarchus labrax*) and the gilthead sea bream (*Sparus aurata*), covering 98% of the national exports (Fig 2). The remaining 2% refers to the sharp snoot seabream (*Diplodus puntazzo*), the red porgy (*Pargus pargus*), the meagre (*Argyrosomus regius*) and other fish species. It is worth mentioning that Greece is the first country to export in EU and the second country considering the whole Mediterranean (Fig 2).

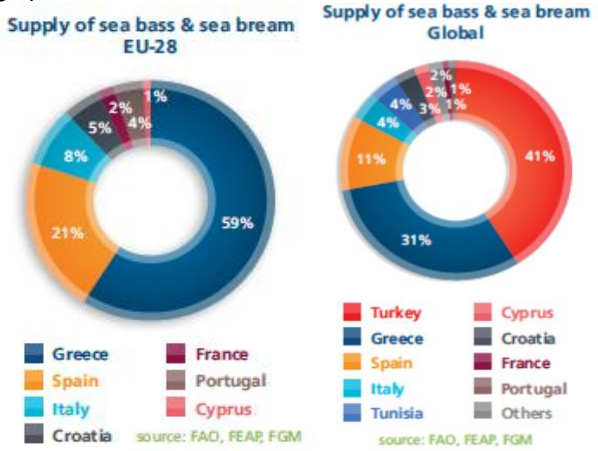


Fig 2. Sea bass and sea bream production in EU and in the Mediterranean

In Greece, the spatial distribution of the fish farms is mostly concentrated into three regions (see Table 1):

1. Peloponnese, Western Greece and Ionian islands
2. Thessaly and Central Greece
3. Aegean

Decentralized Administration	Number of farms	Marine area (acres)	Approved tonnage (%)
Aegean	58	1,135	15.64
Attica	27	486	5.37
Epirus & West Macedonia	41	753	11.53
Thessaly & Central Greece	83	2,037	27.75
Crete	4	75	0.52
Macedonia - Thrace	6	112	1,09
Peloponnese - West Greece & Ionian	117	2,938	38.10
Totals	*336*	*7,535*	*100*

source: FGM, Minagric

Table 1. Spatial distribution of the Aquaculture industry in Greece

Sea bass and sea breams production has increased over the years. In 2016 both species reached 105.000 tons that value almost 553 million euros. Specifically, sea bream production reached 59.000 tons and sea bass production reached 46.000 tons. Other species that represent 2.6% of the total production, reached apx 2.800 tons.

Aquaculture in Greece is constantly growing, with almost 78% of the production exported and 22% distributed on the domestic market. One of the best years, was 2016 since sea-bass and sea bream were exported in 32 countries, according to the National Statistics Agency (ELSTAT), with Italy, France and Spain being the main markets for the Greek products.

According to European Aquaculture Technology Research and Innovation Platform (EATiP), the growth vision predicts annual production increase of 3.1% until 2030 that corresponds to 2.5 million tons of fish. A quite promising prediction is that the Mediterranean aquaculture will rise with 4% annual growth rate that corresponds to 305.000 tons of fish, 1.5 billion euro and 10.000 new jobs. For Greece the relative growth, according to FGM, is shown in Fig 3.

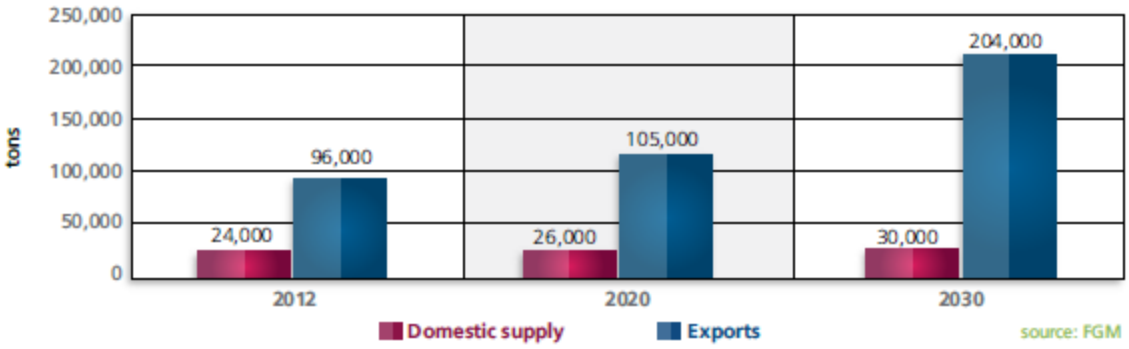


Fig 3. Aquaculture growth vision for the year 2030

1.2 The Biomass Estimation Problem

Effective management in aquaculture farms requires good calculation of fish growth during rearing. Therefore, commercial production and research activities needs for accurate measurement of fish size, requesting for developing technologies to address this problem in a less labor-intensive manner offering improved precision compared to the current approaches.

Today, the most commonly used techniques for length measurement are based on sampling specimens on-site, an approach that requires experienced personnel and induces stress on the fish groups. Additionally, some non-invasive methods have been developed (like the frame in the salmon industry [65]) but they still lack accuracy and are not extensively used. Some experimental approaches using image analysis techniques have also been proposed.

Shortis et al (2013) [9] developed a stereo-video system for measuring fish in aquaculture. It is an automated process that identifies the candidate specimens for measurement, using a specially designed algorithm that measure the fish from snoot to fork, even if their bodies are inclined due to different body orientations. This method was applied in tuna farming.

A semi-automatic fish length estimation system was developed based on stereoscopic computer vision and photogrammetry [60]. This non-invasive method has been developed and tested with salmon. The system consists of two main subsystems, which are a desktop application and a stereoscopic camera. The desktop application is a free software that called VidSync and is operating on Apple Mac OS computers. The use of VidSync software allowed length measurements of manually selected individuals. The accuracy is affected by the quality of the input that the human operator provides. It is quite innovative system but is yet lacking accuracy and is not extremely used. This system has been tested by the HCMR group in the frame of the EU project AquaExcel2020 [59].

Similarly, Linett et al (2001) developed an imaged based system for biomass estimation in free swimming individuals of salmon. The system presented a relative good accuracy with measurement error of 18% and standard deviation of 9%.

Even though these methods still there is no reliable method to estimate the biomass in aquaculture operations allowing for further development. The problem of fish measurements and fish identification is also present in other sectors such as fisheries or ecology. Some of the works done are presented in the following paragraph.

1.2.1 Previous works

In recent years numerous fish size calculation systems have been developed for several applications outside the aquaculture mostly targeting ecological studies and surveys, replacing the standard method for size estimation, based on gathering results on site. As a result, researchers wanted to find a better solution and some of these methods are analyzed below.

In the previous chapter (1.2) methods that are based on stereoscopy are mentioned. Below are presented the methods such as fish size estimation and 3D model. Particularly in [6], [7], [8] different approaches for length estimation are presented, quite innovative methods at this time of period. In the sequence in [21], [22], [23], scientific studies are analyzed methods that referred to three dimensional modeling.

Stereoscopic vision is a method for estimating the size of the samples as well as the distance from the reference camera. A set of cameras combined with a specially designed algorithm could be able to estimate fish's length. A specially designed system based on stereovision, mounted on a common piece of mechanical handling equipment that moves materials from one location to another (conveyor system) could detect and measure fish size from different kinds of flatfish [6], the samples were scanned through a camera. Subsequently an algorithm was detecting fish orientation, length from nose till fork, without being affected from belly flaps. A digital camera received the images and transmitted them into the main system identifying the fish shape, length, orientation as well as its type (flatfish or round fish). The system could process 30000 fish per hour. Fishermen were using it to comply with the EE rules.

Two methods were used to estimate fish length in a work of Rochet et al [7]. The measurements were performed in a water environment with differences from our systems and as in [6] and [8] a completely different procedure was used. The first method refers to a pair of parallel beams, mounted on a specially designed camera, and since the distance of the emissions is known, the necessary measurements are taken. The second method refers to length estimation with an underwater autofocus camera. While there are numerous innovative technologies at this time of period, also the first method had high error rates due to fish movements, while the second method had difficulty detecting the right timestamp during video playback.

In an aquaculture setup however the identification of the candidate fish for measurement represent a second, and even more serious, issue due to the density of the individuals in cages. Some work has been already implemented towards this direction. Several studies were implemented and are presented here that are related to fish recognition. In some studies [21, 22, 23] the authors follow almost the same procedure, building an identification classifier for poisonous fish. D.B. Beeder [21], extracted the characteristics of the samples related to the size and the shape such as length from head to tail and other geometric features. Then a neural network was trained, with 257 frames of which 93 were used for system evaluation. Similar the authors of [22] used the color signature from the ventral part of the fish, in order to extract characteristics based on gray scale to identify optimal features. They trained a neural network that it was able to identify the appropriate fish, based on the characteristics that presented above. The same procedure is followed in [23]. A neural network is again trained based on 47 characteristics per sample, in 4 steps: preprocessing, image processing, segmentation and features extraction. Contrary, in [24], an experiment was conducted and 66 types of characters per sample from different fish species were extracted. Subsequently a sorting tree was built that contained all samples. Each sample was identified and ranked at the database with human intervention. This kind of procedure has 4% better results than the others presented before. The training was held with a dataset of 3179 frames and 10 different kinds of species.
In [25] a system capable of scanning the candidate specimen (e.g fish), extracting all the necessary parameters, was reported. In this work, a signal with bandwidth from 40 kHz to 95 kHz was used in order to scan the candidate specimen, extracting all the appropriate parameters such as, length, height, orientation angle. The work showed clearly the difficulties that could be faced during the implementation of such tasks, but it was very informative because it enhanced with additional details.
The approaches presented before should be further extended and developed taking advantage of the previous works performed in other fields of computer vision and in particular the Particle Swarm Optimization together with artificial intelligence. A similar method is presented in a work by Oikonomidis et al 2011 [17] for tracking the three dimensional position of hand and body poses. The optimization problem was solved using Particle Swarm Optimization algorithm that can adjust on a hand pose even if it has different orientations.

Stereoscopy is based on human optical system, displaying the pictures allows to the human to gain a strong sense of depth. Stereo vision is consisted of two processes, the *display* of the same scene of the two cameras and the *reconstruction* of their three dimensional display. Following the above review of available studies and the studies that are mentioned in chapter 1.2, it is apparent that the stereoscopic vision is the appropriate method for fish measurements, combined with 3D modeling, we will be able to estimate fish's length even if its body was inclined.

1.3 Tools and Methodology

1.3.1 Stereoscopic vision and cameras

Stereoscopic method is based on the human vision system. In order for the method to succeed, cameras take the place of our eyes. A camera is an optical medium which records either static scenes or non-static ones, saving them is a digital medium or in a film. The cameras operate with the light in the visible spectrum. They all operate the same way with the light entering through a converging inclining lens recording an image. Light exposure on the scene is controlled through the shutter. A video camera works in a similar way recording a series of quick succession static scenes. Based on stereoscopic photogrammetry, that a single image can only yield two dimensional coordinates, height and width, with two overlapping images of the same scene, slightly separated from each other, the third dimension, depth, can be estimated. Such systems are based on two cameras while specific software will estimate fish length using algorithms.

1.3.2 Artificial Intelligence and identification with Genetic Algorithms

Artificial Intelligence (AI) is the ability of machinery intelligence. This term refers to machine trying to mimic human cognitive functions such as learning and problem solving. As shown in Fig 4, the current method used was based on human evolution.

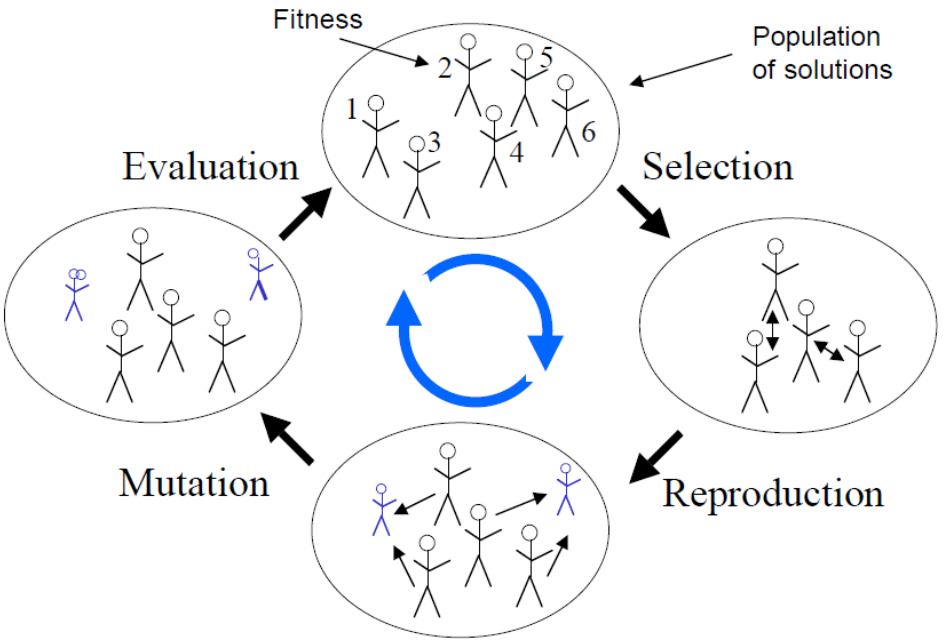


Fig 4. Evolutionary algorithm

The algorithm used was Particle Swarm Optimization (PSO), which optimizes a problem with repeated attempts in order to improve a candidate solution. It uses a population of candidate solutions, where the particles moving in the search area are from known location. This process is repeated until it finds a candidate solution. Analytically PSO is a stochastic optimization technique that was developed around 1995, inspired by the social behavior of fish schooling and bird flocking. PSO has a lot of similar points with genetic algorithms. They both have a population of random solutions, searching for the optimal one for many generations. The solutions that PSO produces are called particles, which are updated in every generation.

Based on previous methods such as body tracking [19, 20] many studies were conducted demonstrating the capabilities of Particle Swarm Optimization as it is hard to measure body poses, due to the many parameters of the body. Furthermore, PSO algorithm was conducted in an even more difficult task such as hand tracking [16, 17, 18]. Studying the methods that referred to artificial intelligence, Particle Swarm Optimization was the candidate method used in this project. Hence, a three dimensional model which through AI could fit on the candidate specimen, was used to measure length and orientation.

1.4 Motivation and Scope of Present Work

Our purpose was to develop a system for fish measurement without any human intervention, while providing accurate measurements. For this stereoscopic photogrammetry was used and a stereoscopic vision system was implemented with a set of web cameras connected to a mini-computer, mounted in a watertight submergible housing.

Artificial Intelligence methods and more specifically Particle Swarm Optimization (PSO) algorithm was used to detected and select the candidate fish for measurement and subsequently the size can be measured through algorithms that employ a three dimensional fish model.

Experiments in a water tank with live fish of known sizes were used to validate the system and assess its accuracy.

Chapter 2: Methodology

As indicated in Fig 5, the developed system for fish size measurement is comprised of three main components, namely the *Stereoscopic Image Acquisition* setup, the *Pre-processing stage* and the *Model fitting algorithm.* Using a stereoscopic vision setup, a sequence of image pairs is taken from two High-Definition IP cameras. The cameras are managed by a mini-computer, synchronizing their scenes and through specially designed algorithms from stereoscopic vision, the system will be able to estimate the depth frame. Subsequently, the system receives the depth frame and through image processing, it is able to isolate potential candidate fish. The image processing involves various stages of filtering to reduce the optical noise from the depth frame due to, e.g., microorganisms and algae. In the final stage, an Artificial Intelligence algorithm, namely *Particle Swarm Optimization* (PSO), will attempt to fit a three dimensional fish model on the candidate sample. When successful, this will allow estimating the length of the candidate fish.

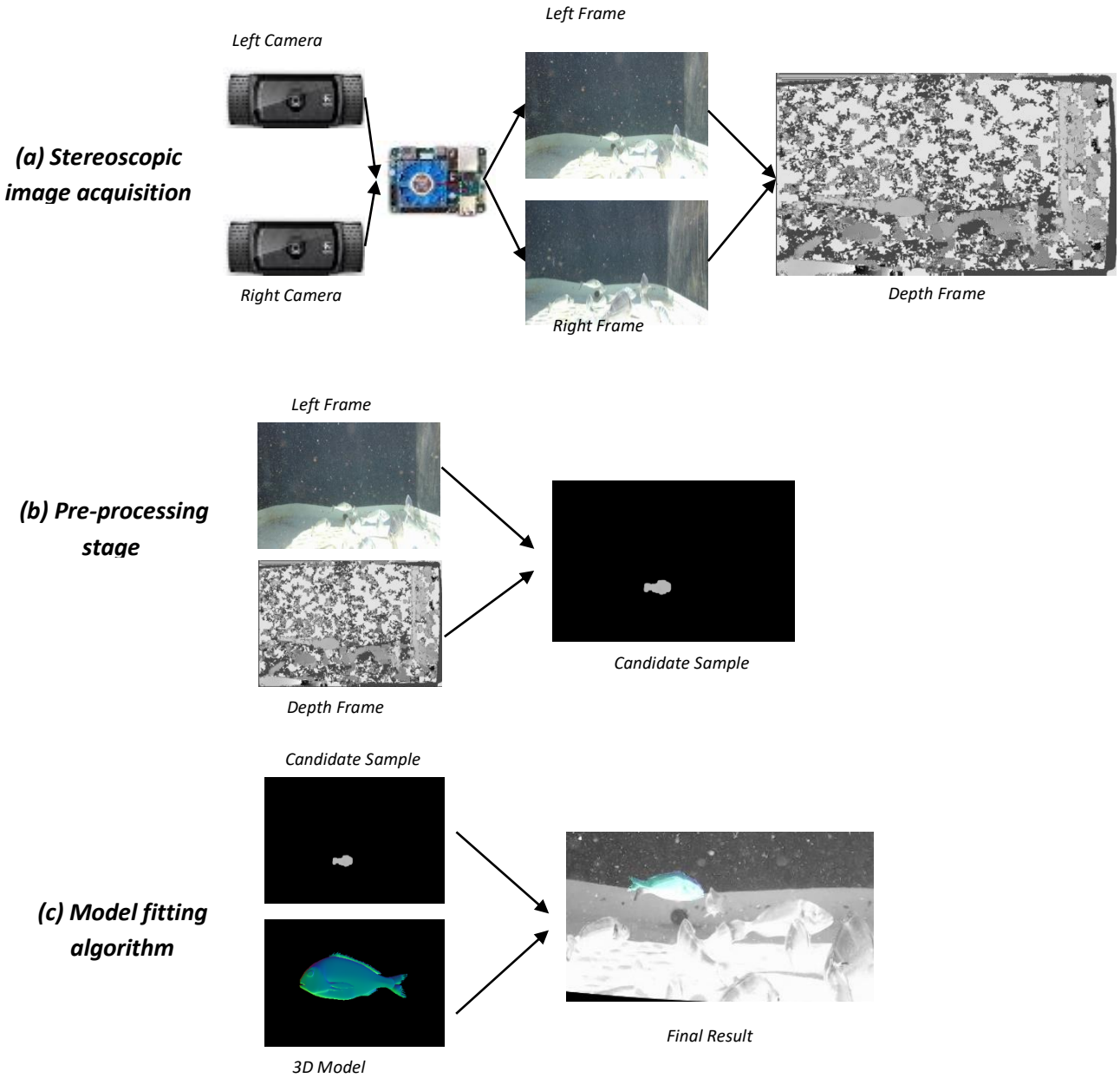


Fig 5. System architecture diagram

2.1 Stereoscopic Image Acquisition

2.1.1 Stereo pair cameras

The system is using two IP cameras (Logitech C920, see Fig 6), providing high resolution (1080p, 30 frames per second), as the better the resolution is, the higher the accuracy of the measurements is. Communication is achieved with a Power-Over-Ethernet link (100 Mbps transfer rate). An additional advantage of these cameras is their small size, easily adaptable to any housing.
In order to be functional the cameras were operated by a mini computer, which receives the data (images/videos) and transmits them into the main system.

Fig 6. IP camera used in the stereo pair

2.1.2 Processor unit

One of the most crucial parts of the system is the Odroid XU4 single board computer. It is a new generation computer with high performance despite being quite small compared to conventional computers. Odroid operates Linux (Fig 7) and offers sufficient storage space and two USB ports, one for each camera. The specifications are represented below:

- Samsung Exynos5422 Cortex-A15 2Ghz and Cortex-A7 Octa core CPUs
- Mali-T628 MP6(OpenGL ES 3.1/2.0/1.1 and OpenCL 1.2 Full profile)
- 2 Gbyte LPDDR3 RAM PoP stacked
- eMMC5.0 HS400 Flash Storage
- 2 x USB 3.0 Host, 1 x USB 2.0 Host
- Gigabit Ethernet port
- HDMI 1.4a for display
- Size: 83 x 58 x 20 mm approx. (excluding cooler)
- Power: 5V/4A.input
- Linux Kernel 4.14 LTS

Fig 7. Single board computer (Odroid XU4)

2.1.3 Waterproof housing

The system ought to be installed in a waterproof housing, protecting it from the harsh aquatic environment. Two special tubes (BlueRobotics SA) (one for each camera) were used, each made of acrylic material and tested in 65 meters depth. Specially crafted holes with appropriate watertight connectors, depicted in the Fig 8, act as entry points for the interconnecting cables.

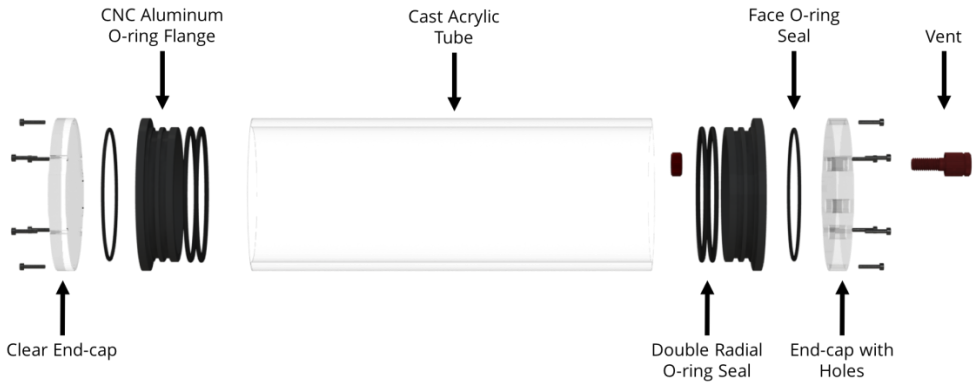


Fig 8. Housing construction

The two tubes were located on an in house build based allowing the adjustment of the distance between the cameras, a feature required for the calibration as explained below. Cameras, minicomputer and POE were installed and adjusted on the bases, maintaining a fixed distance between them (Fig 9).

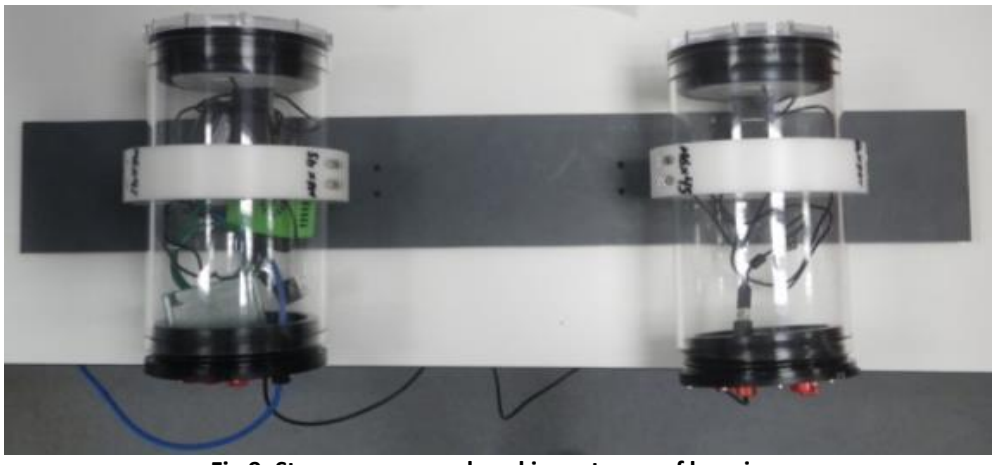


Fig 9. Stereo cameras placed in waterproof housings

2.2 Stereoscopic Vision Module

In order to further analyze the captured image a calibration of the two-camera system is required. The process is explained below.

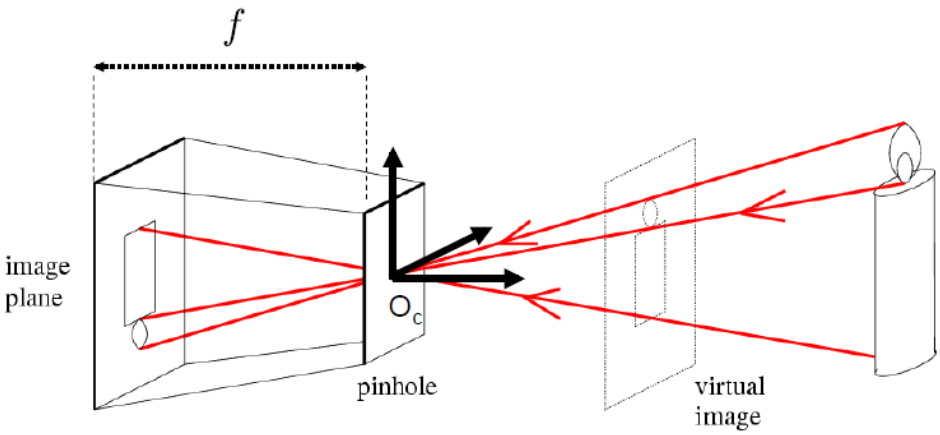
2.2.1 Camera calibration theory

The parameters between an object in 3D space and an object projected in 2D can be estimated through calibration. This is consisting of:

- The Camera intrinsic calibration
- The Camera extrinsic (stereo) calibration

The lens system that any camera is equipped in order to accumulate the reflected light from an object, it also distorts the image as a side-effect.
The two sets of parameters used to create a mathematical model of a 3D camera system are the *intrinsic* and *extrinsic*. The intrinsic parameters are the *Pixel size*, the *Focal length* and the *Principle point* or *Optical center*. The extrinsic parameters are the *Translation* and *Rotation* matrices.

Fig 10 illustrates how the virtual image is depicted through the camera's pinhole. The virtual image is the three dimensional object that through camera's pinhole (O_c – Optical center) is imprinted onto image plane, assuming that $\boldsymbol{f}$ is the camera's focal length.



Source: S Savarese slides.

Fig 10. How image depict through camera's pinhole

To understand how camera's parameters function, it is useful to decompose into a series of operations.

The formula $(\mathbf{1})$ represents the transformation matrix $\mathbf{\Pi}$ where if it will be multiplied by the three dimensional coordinates, it will yield the two dimensional coordinates $\mathbf{x}$.

$$\mathbf{\Pi} = \begin{bmatrix} f_x & 0 & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R_{3x3} & 0_{3x1} \\ 0_{1x3} & 1 \end{bmatrix} \cdot \begin{bmatrix} I_{3x3} & T_{3x1} \\ 0_{1x3} & 1 \end{bmatrix} \quad (\mathbf{1})$$

$\mathbf{\Pi}$ matrix is decomposed into a series of matrices:

The first one stands for intrinsic parameters where f_x, f_y is camera's focal length and C_x, C_y is camera's optical center with respect to the x, y axis.

$$\boldsymbol{A} = \begin{bmatrix} f_x & 0 & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{bmatrix}$$

The second one is the projection matrix, i.e. a unit matrix consisted of aces and zeros.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Finally the third matrix ($\boldsymbol{R}$) is the rotation matrix that rotates the image plane and it's consisted of smaller matrices: The sub matrix $\boldsymbol{R_{3x3}}$ is referred to the image orientation, where a is the orientation angle.

If the angle is indicated for the x axis then the matrix will read $\boldsymbol{R_x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & ca & -sa \\ 0 & sa & ca \end{bmatrix}$, if it is indicated for the y axis then the matrix will read $\boldsymbol{R_y} = \begin{bmatrix} ca & 0 & sa \\ 0 & 1 & 0 \\ -sa & 0 & ca \end{bmatrix}$ and if the angle is indicated for the z axis then the matrix will read $\boldsymbol{R_z} = \begin{bmatrix} ca & -sa & 0 \\ sa & ca & 0 \\ 0 & 0 & 1 \end{bmatrix}$. The ca and sa are actually the $cosine\ (c)$ and $sine(s)$ of the relevant angle.

The 0_{3x1} (matrix of 3 lines and 1 column of zeros) and the 0_{1x3} (matrix of 1 line and 3 columns of zeros):

$$\boldsymbol{R} = \begin{bmatrix} R_{3x3} & 0_{3x1} \\ 0_{1x3} & 1 \end{bmatrix}$$

The last matrix ($\boldsymbol{T}$) is the translation matrix that translates the optical center from the origin of the reference system to the world coordinates, i.e. the 3D coordinates. This again is composed of smaller matrices.

$$\boldsymbol{T} = \begin{bmatrix} I_{3x3} & T_{3x1} \\ 0_{1x3} & 1 \end{bmatrix}$$

The $\boldsymbol{I_{3x3}}$ (matrix of 3 lines and 3 columns) is an identity matrix, a square matrix which all the elements of the principal diagonal are 1 and all other elements are zeros. The 0_{1x3} is similar as above and the sub matrix $\boldsymbol{T_{3x1}} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$ is referred to the translation respectively to x, y and z axis. Analytically T_x is a matrix referred to the translation respectively to x axis, another way of explanation if the translation is respectively to x axis is $T_x = \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix}$, due to the first element of the matrix that corresponds to x axis,in homogeneous coordinates $\begin{bmatrix} x \\ y \\ z \end{bmatrix}$. T_y is a matrix referred to the translation respectively to y axis ($T_y = \begin{bmatrix} 0 \\ T \\ 0 \end{bmatrix}$) and T_z is a matrix referred to the translation respectively to z axis ($T_z = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix}$).

Using equation ($\boldsymbol{2}$), shown below, the three dimensional coordinates are converted into two dimensional coordinates, after multiplying with the transformation matrix. For convenience, the

$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$ coordinates are presented in a matrix $[4x1]$ (4 lines with 1 column) and the $\tilde{\boldsymbol{x}}$ coordinates are presented in a matrix $[3x1]$ (3 lines with 1 column), $\tilde{\boldsymbol{x}} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$.

$$\tilde{\boldsymbol{x}} = \boldsymbol{\Pi} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \boldsymbol{\Pi} \cdot \mathbf{X} \qquad (\mathbf{2})$$

It is useful to note that any point x which belongs to the two dimensional coordinates, can be denoted $\boldsymbol{x} = (x, y) \in R^2$ or $\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$, where R^2 is the two-dimensional space. In order to process the coordinates into matrices, we should translate them into homogeneous coordinates. Homogeneous coordinates are used in computer vision and computer graphics, allowing the description of affine and projective transformations with matrices. Affine transformation is a technique that corrects geometric distortions or non-ideal camera orientations. Essentially it maintains the sets of parallel lines unchanged.

The 2D point can be presented in homogeneous coordinates as $\tilde{\boldsymbol{x}} = (\tilde{x}, \tilde{y}, \tilde{w}) \in P^2$ or $\tilde{\boldsymbol{x}} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{bmatrix}$, where the factors are equivalent differing by a scalar multiplier. Specifically P^2 is called the 2D projective space and is represented by $P^2 = R^3 - (0,0,0)$, where R^3 are the 3 dimensional coordinates.

In order to transform the homogeneous vector $\tilde{\boldsymbol{x}}$ in equation $(\mathbf{3})$ into inhomogeneous vector, it should be divided by the last element ($\tilde{w}$). Then $\tilde{\boldsymbol{x}}$ is called augmented matrix and is useful for the purpose of performing the same elementary row operations.

$$\tilde{\boldsymbol{x}} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{bmatrix} = \tilde{w} \cdot (x, y, 1) = \tilde{w} \cdot \bar{\boldsymbol{x}} \quad (\mathbf{3})$$

where $\bar{\boldsymbol{x}} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ is the augmented vector.

The set of points for which $\tilde{w} = 0$, is called *points at infinity* or *ideal points* and they cannot be converted into inhomogeneous vectors.

Intrinsic parameters extraction

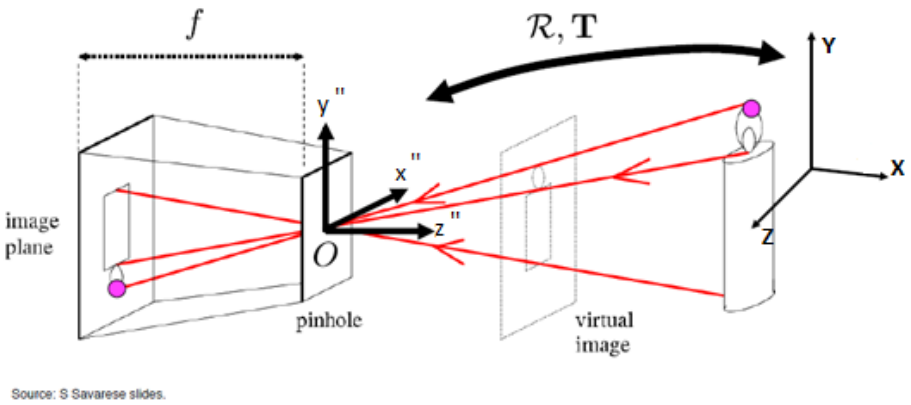
The first step of the calibration process is intrinsic parameters extraction. The intrinsic parameters describe the relation between the coordinates of a three dimensional point and its projection on the image plane. Generally they represent the mapping of a three-dimensional scene to a two-dimensional image and they will help in correcting the distortion caused by the lenses. These parameters do not depend on the projection scene, but on the camera's characteristics that are known to the user, or they can be estimated through calibration by using a well-defined pattern with known distances (e.g. a chess board).

Extrinsic parameters extraction

The extrinsic parameters describe the cameras movement around a static scene or the rigid movement of an object in front of the camera. The Extrinsic parameters are the Rotation matrix (R) and the Translation matrix(T). The extrinsic parameters depend on the position of the camera and represent the transformation of the 3-dimensional world to 3-dimensional camera model.

The 3D point is represented in homogeneous coordinates as $\widetilde{\boldsymbol{X}} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ \widetilde{w} \end{bmatrix} \in P^3$ or into inhomogeneous coordinates as $\boldsymbol{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \in R^3$, where P^3 is the three dimensional projective space and R^3 is the three dimensional coordinates. It is useful to denote that for the following calculations a 3D point using the augmented vector $\overline{\boldsymbol{X}} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$ with $\tilde{X} = \widetilde{w} \cdot \bar{X}$.

In the example given below (Fig 11), the coordinates of an object in 3D space will be translated into a coordinate system related to the camera.



Source: S Savarese slides.

Fig 11. Point translation from 2D coordinates to 3D coordinates

The three dimensional coordinates of the object in the system $\boldsymbol{X}$ need to be translated into the two dimensional coordinates in the system $\boldsymbol{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$.

Initially the extrinsic parameters will be multiplied with the three dimensional coordinates $\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$, to yield the point $\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix}$ with respect to the camera (equation $(\mathbf{4})$). The conversion below is equivalent if $Z \neq 0$.

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \boldsymbol{R} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \boldsymbol{T} \quad (\mathbf{4})$$

Subsequently, since the intrinsic parameters are known, we are able to estimate the 2-dimensional coordinates on the image plane, from equations $(\mathbf{5})$ and $(\mathbf{6})$ below:

$$x = f_x \cdot x'' + C_x \quad (\mathbf{5})$$

$$y = f_y \cdot y'' + C_y \quad (\mathbf{6})$$

The sum up of the above calculations into a single equation is shown in equation $(\mathbf{7})$ below:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \boldsymbol{A} \cdot (\boldsymbol{R}|\boldsymbol{T}) \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (7)$$

Image Rectification

Following the Intrinsic and Extrinsic parameters calculation, and in order to validate the initial estimations, the correlation of the correspondence points takes place. This is performed through image rectification, a transformation process that is used for frame projection into a common image plane, using two images (from Left-hand and Right-hand cameras) that depict an object from different viewpoints and from known relative camera positions. Each pixel from the left image must correspond to one pixel from the right image, and vice versa. Analytically, a sufficient number of images from a specially designed pattern (in our case a chessboard pattern), recorded from different angles and orientations is used, and afterwards, the corresponding points are matched using epipolar geometry.

In Fig 12 below the principle of epilpolar geometry is shown, where O_L and O_R represent the Left-hand and Right-hand cameras, respectively.

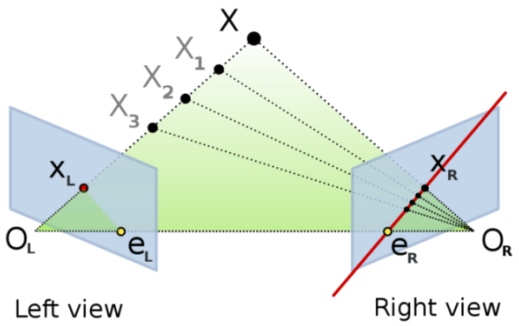


Fig 12. Epipolar geometry

Assuming that the point X_L is known and the difference between $e_R - X_R$ is also known, the point X_R is the projection of the point X onto the right image plane. It means that the point that displayed onto the right image should be also displayed onto the left image. This provides the epipolar constraint that this point should be contained in the geometry section $e_R - X_R$. This point will be validated from the points that lay on the line $O_L - X_L$.

Another way to explain the above, is by using the essential and fundamental matrices between cameras (Fig.12). If the image planes are parallel to each other, the distance between the cameras is constant, and their optical centres are on the same horizontal plane, the focal lengths are also the same, then the epipolar lines are parallel to horizontal scan lines oriented to infinity $\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$, with respect to the x axis, due to the first element of the matrix that corresponds to $\begin{bmatrix} x \\ y \\ z \end{bmatrix}$, in homogeneous coordinates.

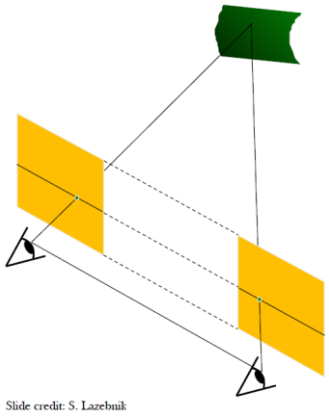


Fig 13. Rectified images

In order to rectify stereo images (gray frames) (Fig 14), the image planes must be reprojected onto a common plane parallel to the optical centers (yellow frames - Fig 13). This transformation is achieved from the formula $(\mathbf{8})$ below.

$$\boldsymbol{x}^T \cdot \boldsymbol{E} \cdot \boldsymbol{x}' = 0 \quad (\mathbf{8})$$

Where $\boldsymbol{x}$ is referred to the left camera's coordinates and $\boldsymbol{x}'$ is referred to the right camera's coordinates. The superscript T in the variable $\boldsymbol{x}^T$ refers to the transpose of the $\boldsymbol{x}$, (i.e. the rows become columns and vice versa). For example if the point x is $\boldsymbol{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ then $\boldsymbol{x}^T = [x \quad y \quad 1]$.

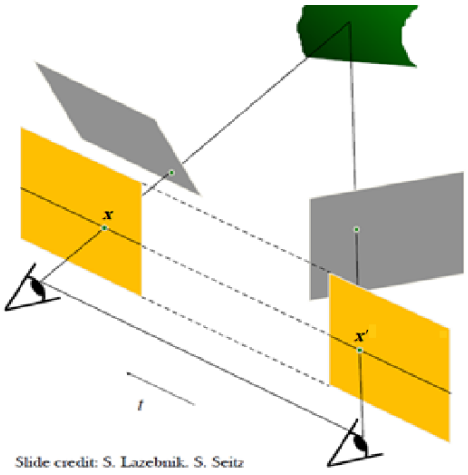


Fig 14. Grey images before rectification

The matrix E in equation $(\mathbf{8})$ is a joint matrix of rotation R and translation T relatively to x axis, where T_x is the translation of x axis.

$$\boldsymbol{E} = [T_x] \cdot \boldsymbol{R} \quad (\mathbf{9})$$

The matrix of E can be written analytically in equation $(\mathbf{10})$:

$$\boldsymbol{E} = [T_x] \cdot \boldsymbol{R} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \qquad (\mathbf{10})$$

Based on the Fig 14, to evaluate the rectification of the yellow images, they must satisfy the following constraints. The corresponding point[1] should have the same vertical coordinates (relatively to y axis) or the epipolar lines should be parallel to the horizontal axis x. Equation $(\mathbf{11})$ demonstrates that the corresponding points of the y coordinates are similar. It should be noted that the symbols x and x' are refer to the points of the two yellow images. More specifically, the coordinates of the point x (left frame) are x_x and x_y and the coordinates of the point x' (right frame) are x'_x and x'_y.

$$[x_x \quad x_y \quad 1] \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \cdot \begin{bmatrix} x'_x \\ x'_y \\ 1 \end{bmatrix} = 0 \rightarrow [x_x \quad x_y \quad 1] \cdot \begin{bmatrix} 0 \\ -T \\ T_{x'_y} \end{bmatrix} = 0 \rightarrow T_{x_y} = T_{x'_y} \quad (\mathbf{11})$$

As a conclusion the final result $T_{x_y} = T_{x'_y}$ points out that the translation of the x point of the left camera is equal to the translation of the x' point of the right camera.

Depth Estimation

After we have calculated the transformation matrix (intrinsic & extrinsic parameters) and the rectification of the images have been performed, the final step is the depth estimation. Depth is the distance of an object from the reference camera and in order to calculate it, the disparity estimation should take place first. Disparity is the difference between the two cameras relatively to x axis, and it is calculated using triangulation.

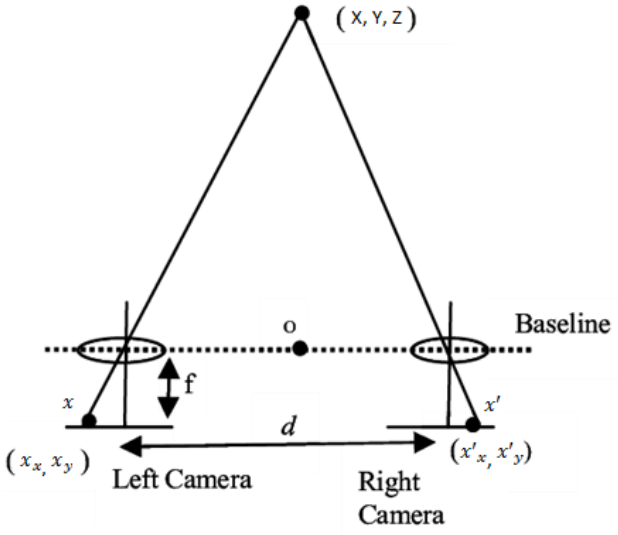


Fig 15. Two rectified scenes projecting same plane and matching corresponding points

As Fig 15 shows, assuming that the points x and x' are known, then their projection lines are also known. Provided that the points correspond to the same three dimensional point (X, Y, Z), means that their projection lines should be intersect to the point (X, Y, Z). Through the triangulation

[1] Two points, one in each frame, which when correlate they give the same perception of the image,

method the point (X, Y, Z) can be calculated. As we mentioned before disparity is the difference between x_x and x'_x (equation($\mathbf{12}$)).

$$Disparity = x_x - x'_x \quad (\mathbf{12})$$

Subsequently, since the baseline B (distance between two cameras) and the focal length f (an intrinsic parameter) is known, the Depth can be calculated using equation($\mathbf{13}$) :

$$Depth = \frac{B \cdot f}{Disparity} \quad (\mathbf{13})$$

A specially designed algorithm was able to estimate the depth in any frame and following this to estimate a ”depth frame”.

2.2.2 Synchronization

An initial stage before the calibration is the synchronization of the scenes, a crucial step for the parameters calculation. In order for a system to operate in unison, the coordination of events was required. In our case, we had to synchronize the received images having common the correct time and the correct frame. This was carried out by developing a code based on opencv libraries, resulting in successful frame synchronization. As the system frame rate was 20 frames per second, each frame requires 0.05 seconds to be sent. Hence a special algorithm was developed to evaluate the intermediate time between commands, calculating the delay (Fig 16). The highest delay between commands was t = 32630 μsec. It can be concluded that the system can properly sustain the prescribed frame rate without incurring additional delays.

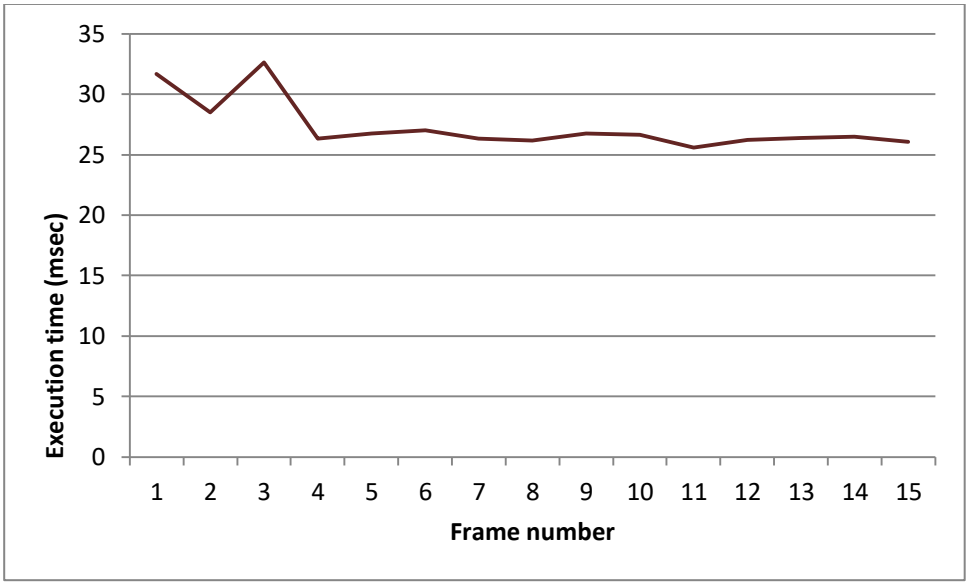


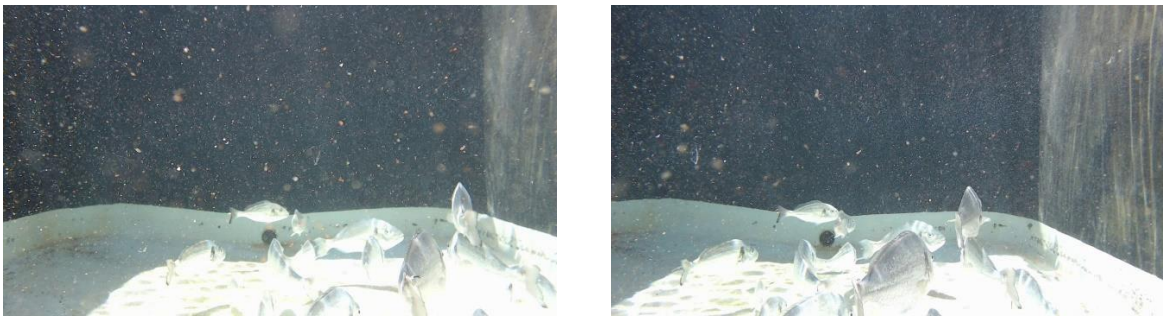
Fig 16. Evaluate synchronization

2.3 Pre-Processing

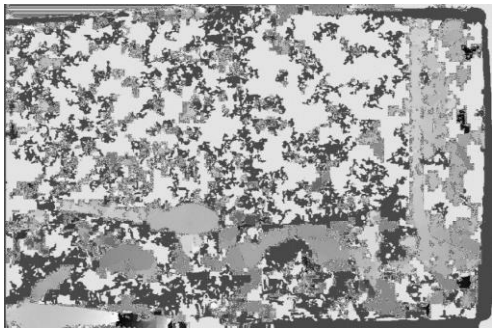
Based on quantization, a process of separating the data into discrete sets, depth frame was separated at intermediate lower depth values. The system processes the various depths, detecting all the candidate fish in the frame. Based on mathematical morphology we were able to fix up the image, by two basic operations, namely *erosion* and *dilation*:

- **Erosion**: In order to find the appropriate fish samples, we had to deal with the extra information, blurring the picture due to algae, microorganisms, remaining foods. As it was mentioned before through discrimination of depth values, the depth image was separated into specific amplitudes. The problem was that the microorganisms were depicted on the image, affecting the finding of the candidate fish. So by filtering the image with the erosion method, the small areas of noise were eliminated, keeping the appropriate areas of the candidate specimens. What is worth mentioning however is that the fish areas on the image were also adversely affected. All the pixels that were near the threshold were discarded, depending of the kernel size.

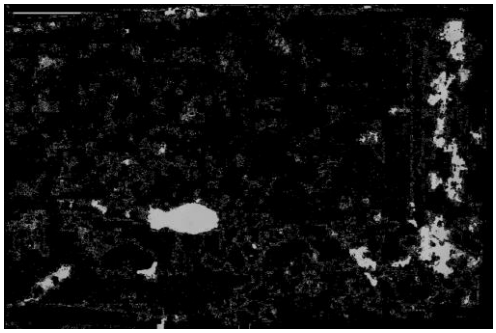
- **Dilation**: The inverse of erosion is called dilation and it expands the shapes. Helps to expand the connected set of ones in the binary image. With all the white regions increased. Due to erosion, that removes white noises and shrinks our object, its original state had to be assumed.



(a)



(b)



(c)

Fig 17. (a) Sample left and right frames acquired by the system. (b) Extracted depth map before image processing. (c) Depth map after image processing.

In Fig 17 (b), the depth map, extracted from the stereo pair of Fig 17 (a) is shown without any processing, while Fig 17 (c) depicts depth after processing. To avoid detecting regions with more than one fish, the overlapping window's amplitude method was applied. During image processing a range of window's amplitude was set, as well as an additional range for the overlapping window. For instance, if the system was searching in a depth range between 150 cm-160 cm, then three overlapping windows (images) are created:

- 1st Image range: 145cm-155cm

- 2nd Image range: 150cm-160cm
- 3rd Image range: 155cm-165cm

The remaining noise during image processing would make the system more difficult to fit the 3D model. Removing noise would help the system to eliminate inappropriate regions. In order to calculate noise regions, a minimum depth value was set (determined by the reference camera) as well as a maximum depth value (tank length). In turn, the appropriate components (fish) were manually selected, erasing the remaining from the frame as noise. This procedure was carried out in a large number of frames (depicted in the histogram below Fig 18), exporting the maximum noise regions from a number of images.

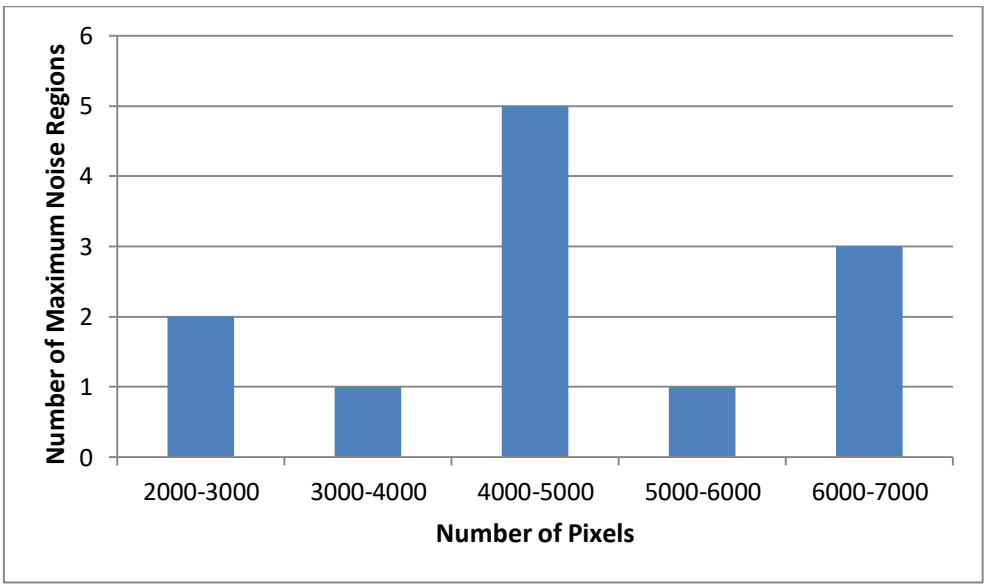


Fig 18. Noise components diagram

The highest noise region rose up to 6521 (number of pixels) and the smallest specimen (20 cm) was detected at a distance of 190 cm. The specimen's region was measured and it was 5760 (number of pixels), based on this ascertainment, a threshold for inappropriate noise elimination was set up to 5700 (Fig 19).

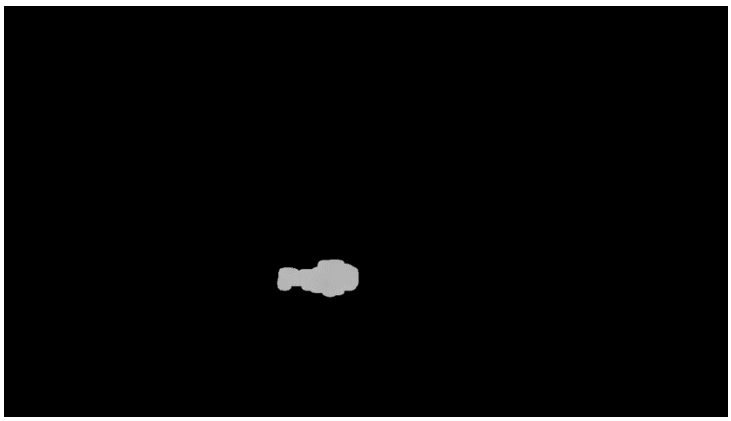


Fig 19. After setting up noise threshold

This would provide the appropriate specimen. Following a series of measurements from the available data sets a threshold was defined at 5700 pixels, as it was observed that the system could detect the smallest specimen (20cm) in the distance of 190cm. Based on these measurements a minimum range was applied at 140 cm, in order to detect all fish sizes.

As the algorithm succeeds in detecting the appropriate specimen, the coordinates of the candidate's region become available. So the 3D model could fit through the PSO algorithm, giving us the specimen's length.

2.4 Model Fitting Algorithm

3D Model

Gilthead Sea Bream (*Sparus aurata*), one of the most important commercial aquaculture fish species, was used as a model species. In Fig 20 a graphical representation is shown.

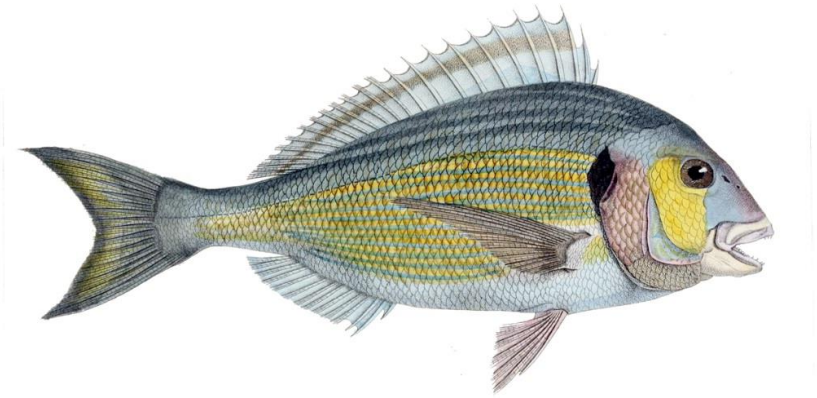


Fig 20. Sparus aurata - Gilthead Sea Bream

A three dimensional model was developed, representing the target fish (Fig 21). It was designed using the *Blender* open source software.

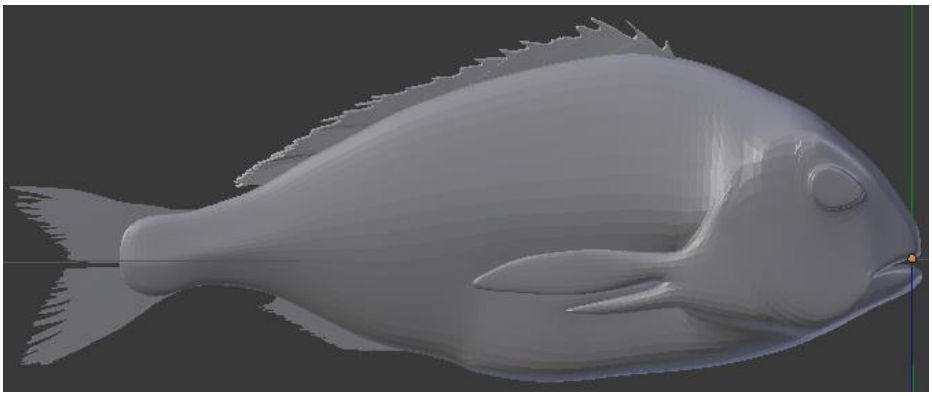


Fig 21. Designed fish without skeleton

The objective was to create a model with precision as to allow "natural" movements and "positions" depicting all the parts of the body. The model was processed to adapt to the different sizes and orientations of the candidate fish. Based on that, a skeleton (Fig 24) was considered, consisting of joints and links:

- Joints: Each joint has 3 degrees of freedom, achieving the desired link position in the world space.
- Links: It is a body of solid material that connects two adjacent joints.

The skeleton determines the orientation of the model.

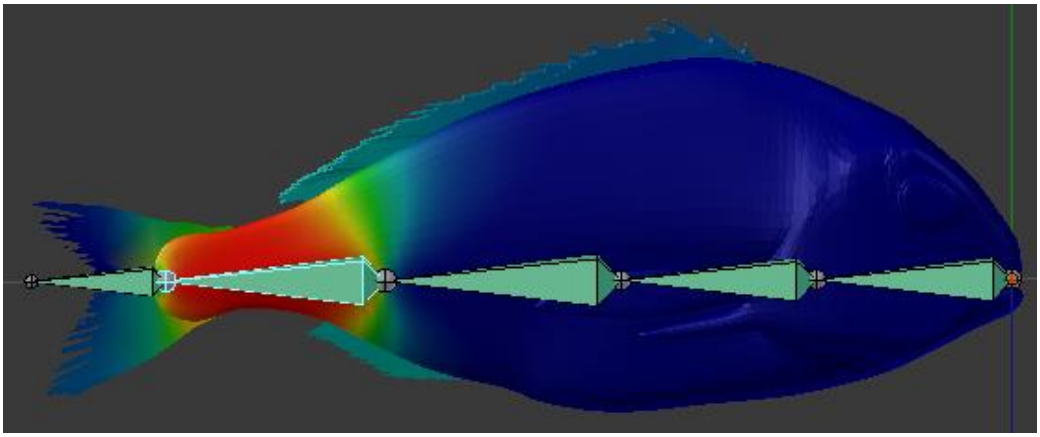


Fig 22. Designed fish with skeleton

As the skeleton is on scale, the model is adjusting appropriately on the candidate fish. In Fig 22, some parts on the model present different color variations. As the color gets “warmer”, the allowed movement range in this area is increasing. Conversely, areas whose color is “cold” (i.e, towards blue) have a decreased movement range. All marine fish do not have the same movement mode. Based on bibliographic references [3], gilthead sea bream belongs to the family of carangiforms (Fig 23), which means that it uses only approximately 1/3 of its body length during swimming motions.

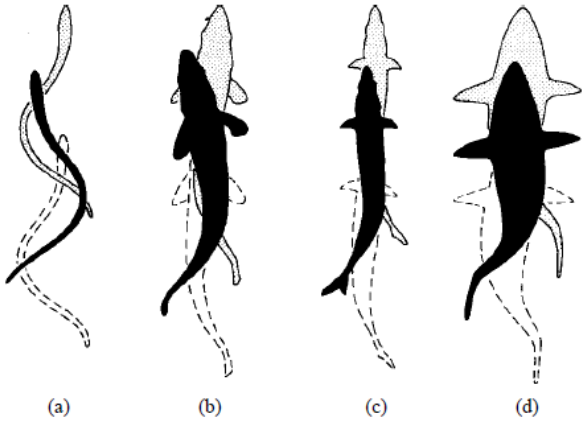


Fig 23. Swimming movements (a) anguilliform, (b) subcarangiform, (c) carangiform, (d) thunniform mode

Hence, the skeleton backbone was edited, placing the appropriate constraints on the links and joints (Fig 24).

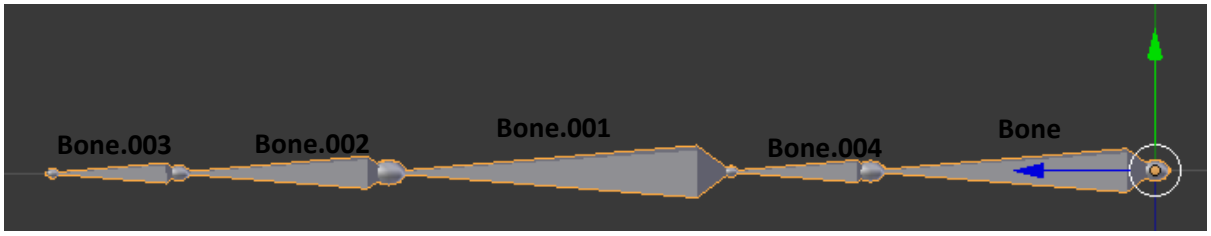


Fig 24. Defining the parameters on the skeleton

A skeleton is composed of 5 joints and 5 links and is adjusted in a way that it can represent the shape of the candidate fish during any swimming motion. The specific constraints of each joint are presented below.

- Skeleton length (in our case it was defined 2,5 cm)
- Skeleton scale: The scale is adjustable, in order to fit all candidate fish's lengths. The range of the scale is between 0,1 and 50.
- Skeleton orientation (Angle maximum range per joint):
  - Bone.001: X → 0°, Y → 0°, Z → (-15° - 15°)
  - Bone.002: X → 0°, Y → 0°, Z → (-30° - 30°)
  - Bone.003: X → 0°, Y → 0°, Z → (-40° - 40°)
  - Bone.004 : X → 0°, Y → 0°, Z → 0°
  - Bone: X → 0°, Y → 0°, Z → 0°

The final model used consists of 11 parameters:

- 3 parameters that refer to model translation in the 3D system (X, Y, Z) axes).
- 4 parameters for model orientation (quaternion representation), in order to achieve the desired body orientation.
- 1 parameter for model scale, to adjust properly all lengths of fish.
- 3 parameters for each joint orientation (Euler angles), adjusting all fish's body inclinations.

Particle Swarm Optimization algorithm

As it has been mentioned earlier, the Optimization of 3D model fitting is achieved using the Particle Swarm Optimization (PSO) algorithm, which optimizes a problem by making repeated attempts to improve a candidate solution. In this case, it emulates the interaction of 3D model's particles (potential solutions) through a number of generations. Each next-generation's particle is optimized over the current-generation's optimum particles in the search space. The optimal results are used in the next phase, until the number of generations (iterations) reaches the end. The total number of generations, i.e. the number of algorithm's iterations, is determined by the user. A good example is the evolution of the human species (Fig.25).

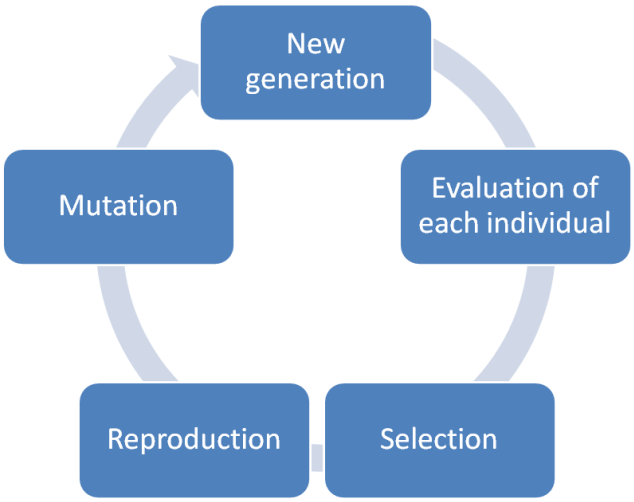


Fig 25. Evolution algorithms

To optimize the population (the number of particles) that lies in the parameter space, the use of an *objective function* is required, having as goal to evaluate all candidate solutions. In each generation, every particle is characterized by its current *position* and its current *velocity*. The i_{th} particle stores the optimal position that comes from the evaluation of the objective function. When the best

position is achieved all the particles of the swarm occupy the model, which is called global optimum. This technique is similar to the way ants find the optimal route between their nest and a food source as illustrated in Fig 26. In the first figure, two ants follow different paths to find food. In the second figure both ants evaluate the routes that they followed, in order to find the optimal. Subsequently in the third figure all ants have been informed about the optimal route.

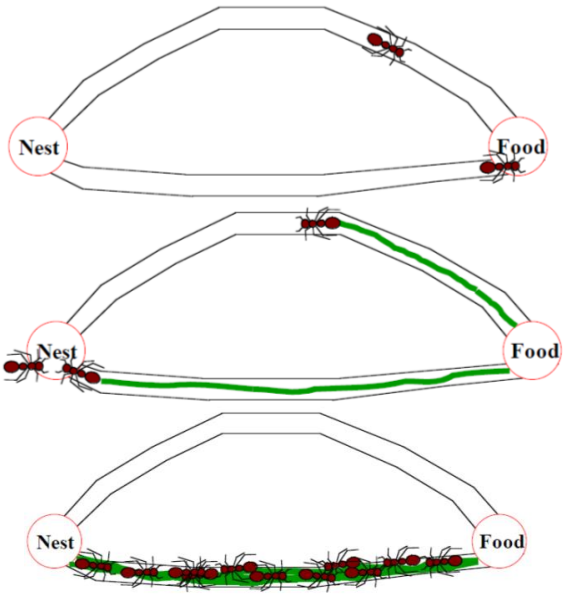


Fig 26. Ants searching for food, finding optimal route

In our case, to achieve an acceptable 3D model fitting using PSO, each particle decides the way that it will be directed, based on previous searches and the space that is placed. It tries to find the best solution, continuously assessing previous results. Each particle stores the best value (highest fitness). The PSO algorithm uses some parameters that help in its evolution, such as:

- p_{best} : Each particle keeps track of its coordinates in the search space which are associated with the best solution (fitness) it has achieved so far.
- l_{best} : It is the best value obtained so far by any particle in the neighbors of the particle.
- g_{best}:The set of the swarm particles with the optimal values that belong into a specific space, after they have been evaluated by their p_{best} and l_{best} values [64] :

With the parameters defined in the previous sections for the skeleton, particles manage to adapt on the candidate sample, evaluating every time the candidate solutions, separating the predominant one. The final goal is the matching of the 3D model on the candidate specimen.
To achieve this, a parametric model of joint kinematics of a fish was defined. As already mentioned, it has 11 parameters that represent the fish pose (3-D position and 4-D quaternion encoded orientation). Therefore, mapping to the feature space is required for each point in search. In our case, to achieve the appropriate fitting we relied on these parameters:

- Candidate fish's observation
- Depth map

By using different values for the parameters, the model is evaluated in order to examine if it can yield all the orientations with the given constraints. In this way the estimation of the fish length even

when the body is bent can be performed. Fig 27 below shows how the system visualizes the fish model.

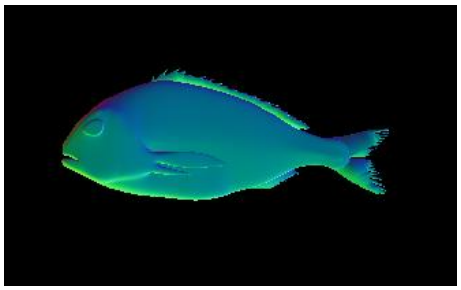


Fig 27. 3D model visualization

All of the model's parameters are presented below:

- x_t : Particle's current position
- u_t : Particle's current velocity
- t : Generation number
- P_g : Global best
- K : Constant constriction factor (value calculated for the 3D model used in each application [43])

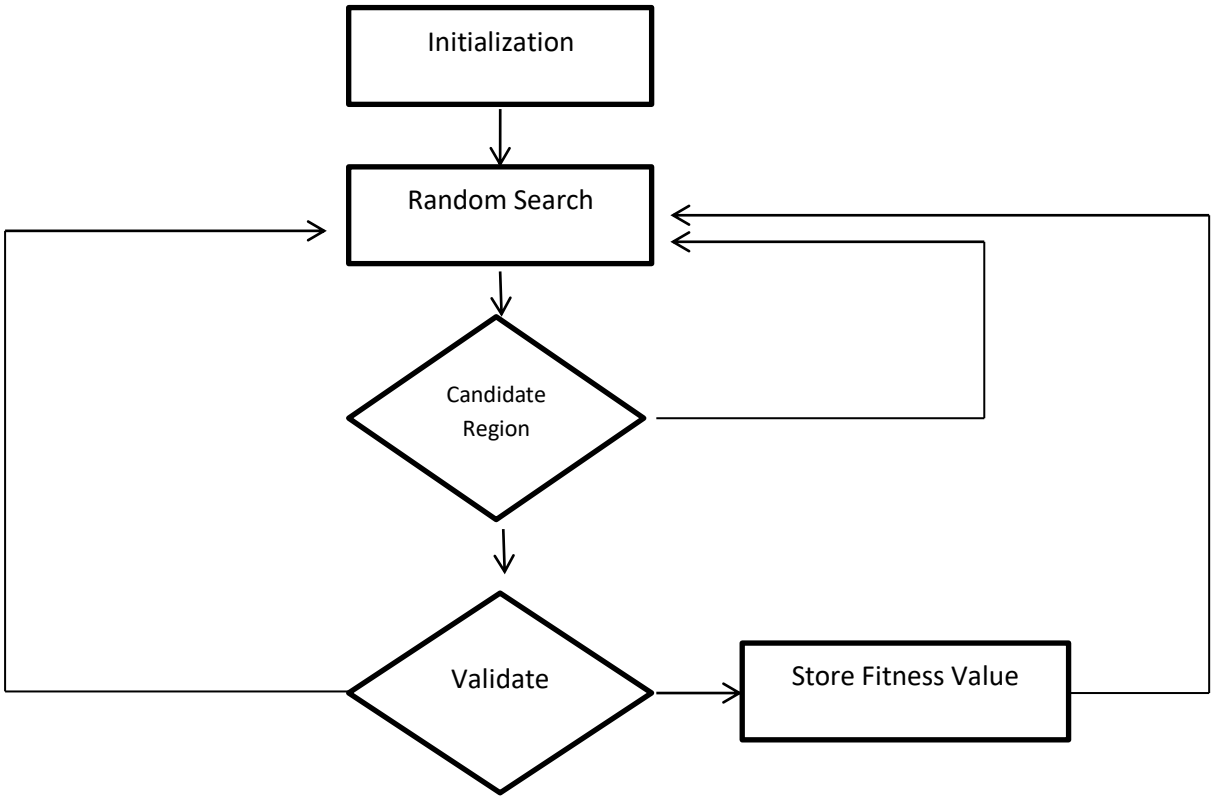


Diagram 1. Implementation diagram of the PSO algorithm

A detailed description of the PSO algorithmic steps (Diagram 1) is provided next.

1. Initialize a population array of particles with random positions and velocities on D dimensions in the search space.
2. **Loop**
    a. For each particle, evaluate the desired optimization fitness function in D variables.
    b. Compare particle's fitness evaluation with its p_{best}. If current value is better than p_{best}, then set p_{best} equal to the current value, and $\overrightarrow{p_i}$ equal to the current location $\overrightarrow{x_i}$ in D-dimensional space.
    c. Identify the particle in the neighborhood with the best success so far, and assign its index to the variable g.
    d. Change the velocity and position of the particle according to the following equation:

$$\overrightarrow{u_i} \leftarrow \overrightarrow{u_i} + \vec{U}(0, \varphi_1) \otimes (\overrightarrow{p_i} - \overrightarrow{x_i}) + \vec{U}(0, \varphi_2) \otimes (\overrightarrow{p_g} - \overrightarrow{x_i})\overrightarrow{x_i} \leftarrow \overrightarrow{x_i} + \overrightarrow{u_i}$$

    e. If a criterion is met (usually a sufficiently good fitness of a maximum number of iterations), exit loop.
    f. End loop

Notes:
– $U[0, \varphi_i]$ represents a vector of random numbers uniformly distributed in $[0, \varphi_i]$ which is randomly generated at each iteration and for each particle.
– $\otimes$ is component-wise multiplication.
– In the original version of PSO, each component of v_i is kept within the range [−*V*max*,* +*V*max]

As mentioned, PSO algorithm will optimize the 3D fish model on the candidate specimen. It simulates the "social interaction" of a population of particles whereby a number of generations are evolved. Every particle holds its initial position in a vector x_t set of parameters and its current velocity in a vector u_t. The i_{th} particle stores its position in a vector p_i, corresponding the best optimal evaluation of its objective function up to the current generation t. The particles store the best position p_g results in a vector which corresponds to the best evaluation until they reach all the generation.
In each generation t, the velocity changes according to equation $(\mathbf{14})$

$$u_t = K\left(u_{t-1} + c_1 r_1 (p_i - x_{t-1}) + c_2 r_2 (p_g - x_{t-1})\right) \quad (\mathbf{14})$$

and its position according to equation $(\mathbf{15})$

$$x_t = x_{t-1} + u_t \quad (\mathbf{15})$$

In the above equations $(\mathbf{14})$, K is a constant *constriction factor*, c_1, $c_2$are the *social components,* while r_1 and r_2 are random samples of a uniform distribution in [0..1]. The algorithm has a search space and it is a multidimensional cuboid. Initially the particle velocities are set to zero and the particle positions are randomly initialized. During optimization procedure, if velocity forces particles to move outside the search space, then this particle value is set to zero and does not perform any move at the corresponding dimensions. The final outcome of PSO is $P_*$with the best score across all generations. The search space of the fish 3D model is a parameter space m. The fish model is represented by 11 parameters, hence the search space has 11 dimensions. The population is a set

consisting of the hypothesis of the 3D fish-object configurations and the objective function minimized by $O(m, M)$. The results from PSO

$$p_* = m_* = argmin_m\big(O(m, M)\big) \qquad (\mathbf{16})$$

represents the best guess of the algorithm for the joint fish-object model parameters m given the multiframe M.
The objective function $O(m, M)$ depends on the compatibility of 3D model's observation with the characteristics of the image. The objective function is defined as:

$$O(m, M) = \sum_{I \in M} D(I, m) + \lambda_\kappa W(m)$$

In the above expression the first term quantifies the discrepancies of the fish model m to the observations that based on the camera, while the second term quantifies the penetration depth between the fish and the object, with λ_κ being a weighting factor [18].

At the beginning, the fish model is manually placed at a roughly predefined position and pose. Worth noting is that two optimizations are taken for each object, one with a starting point from the nose and one with a starting point from the tail. In this way we are able to detect fish orientation and its length, evaluating the results from evaluation function, the result of the evaluation function is the score and denoted as $\boldsymbol{\Omega}$. The solution for multiframe M_{t-1} is used to bootstrap the initial population for the optimization problem M_+. The multiframe $M_+$is performed for a fixed number of particles and generations. The first member of population m_{ref} for M_t is the solution for M_{t-1}. The best hypothesis m_* is dubbed as the solutions for time step t [16,17,18].

2.5 Length Estimation

In order to calculate the candidate's specimen length, the value of the optimizer's scale should be translated into world scale, a process called *normalization*. Consider I is the value that gives the optimizer scale. In each measurement the values should be normalized into the range $0.1 \leq I \leq 50$. Where I_n is normalization value. So we have two equations with two unknown values a and b:

$$I_n = a \cdot I + b \qquad (\mathbf{17})$$

Hence, $I = I_{min}$ the results are $0.1 = a \cdot I_{min} + b$
$I = I_{max}$ the results are $50 = a \cdot I_{max} + b$

Based on the above equations and since $I_{min} = 0, \quad I_{max} = 1$, the unknown parameters are calculated as $a = 49.9$ and $b = 0.1$ to finally yield:

$$I_n = +49.9 \cdot I + 0.1 \qquad (\mathbf{18})$$

Chapter 3: Accuracy Assessment

The verification of the system started by its calibration and then through experiments involving the measurement of live fish of known sizes.

3.1 Camera Calibration in Tanks

An algorithm was developed to detect a special pattern on the chessboard, with given parameters:

- Finding pattern
- Known distance between corners.

The algorithm ought to recognize the given pattern on a chessboard at different distances and orientations, estimating intrinsic parameters and correcting the distortion from lenses. In Fig 28 the right frame shows the image after a distortion correction, contrary to the left frame which depicts the image with distortion.

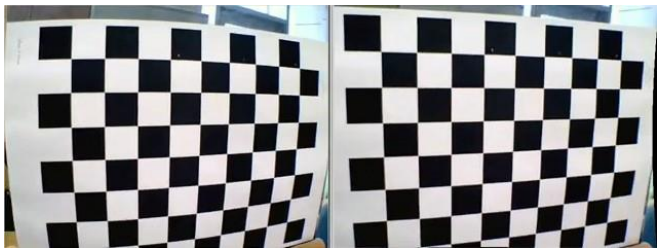


Fig 28. Camera calibration and distortion correction

After intrinsic parameters were estimated for each camera separately, the same procedure was followed for both cameras simultaneously. Rectification has been achieved in calibration images (Fig 29) after estimating the transformation matrix (translation and orientation).

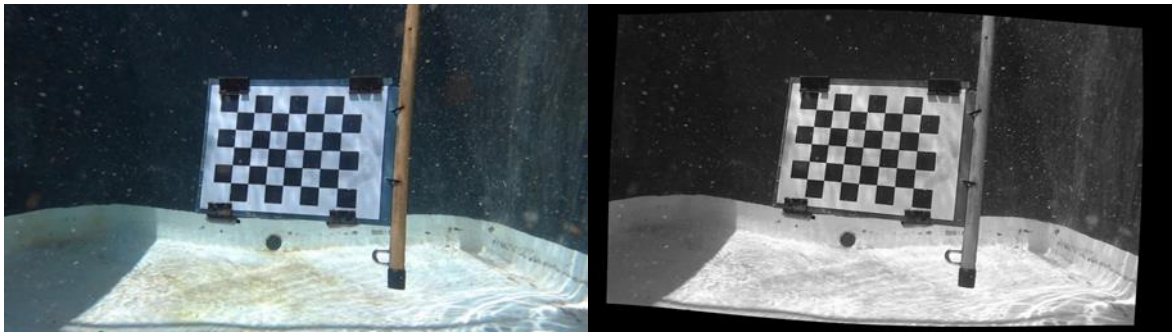


Fig 29. Left frame - unrectified image, right frame - rectified image

To verify if scenes are correctly rectified epipolar lines were outlined to evaluate (through visual inspection) if the chosen random points instersect from the same line (Fig 30).

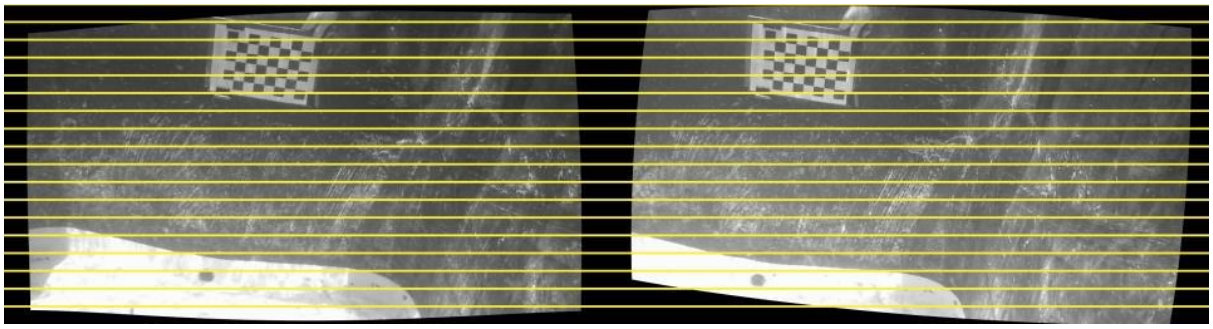


Fig 30. Epipolar lines showing proper calibration

Knowing the disparity map and applying simple trigonometry, the formula of depth was used as shown in paragraph 2.2.1:

$$Z = \frac{B \cdot f}{\text{Disparity}} \qquad (\mathbf{13})$$

Where f is the focal length (previouly estimated from intrinsic parameters) and B express the distance between two cameras (baseline).

Accepting that the baseline is proportional to depth, when the distance between the cameras is increasing at the same time the field of depth is also increasing. Supposing an object is close to the cameras, then the overlap between the image field is reduced, making difficult to correlate points. Due to the results that depicted in Fig 31 and Fig 32, the distance between the cameras (baseline) was adjusted to 19 centimeters.

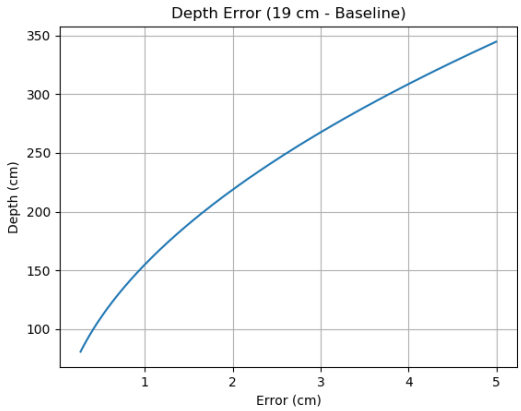


Fig 31. Depth - error diagram

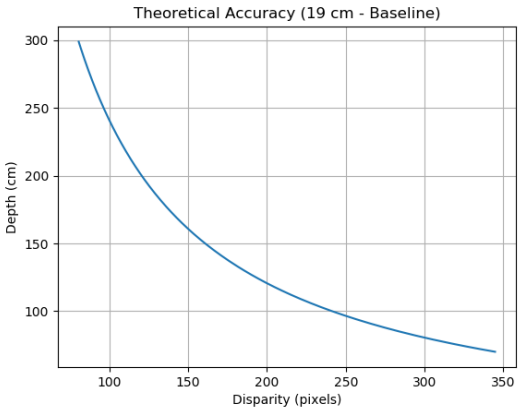


Fig 32. Diagram of theoretical accuracy (19 cm - Baseline)

Based on equation $(\mathbf{13})$ and given the disparity map (Fig 33), the baseline and focal length, depth can be exported. As mentioned in chapter 2.2.1, a specially designed algorithm was able to estimate the depth in any frame and following this to estimate a ”depth frame”. Depth mapping is shown in Fig 34.

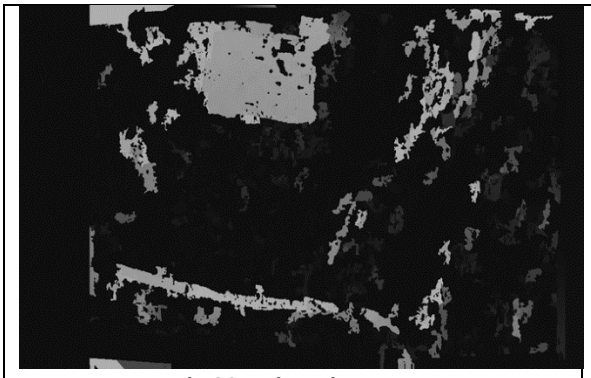


Fig 33. Disparity map

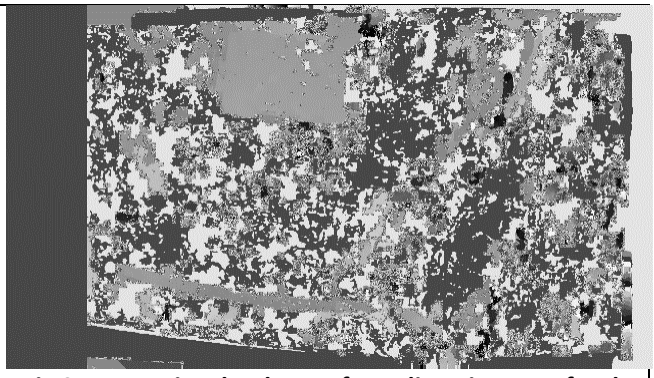


Fig 34. Exporting depth map from disparity map, focal length = 1270px, baseline = 19cm

3.2 Validation

A first validation of the system by measuring known objects took place in lab environment (Fig 35) where known objects (chessboard, cup, batteries) were measured.

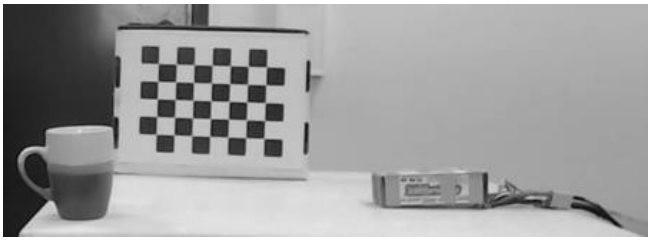


Fig 35. Measurement of known objects

In the diagram (Fig 36) the results of this first evaluation are shown.

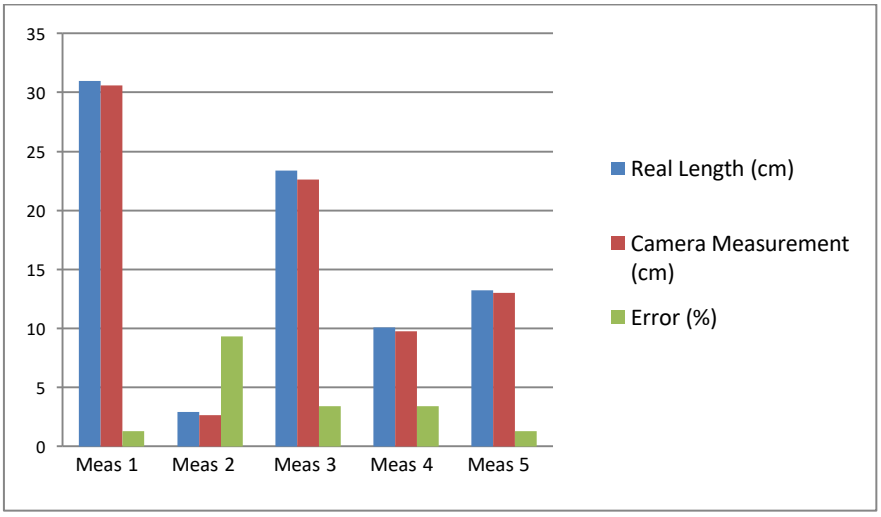


Fig 36. Measurements taken outside from the aquatic environment

For each object two arbitrary points were selected manually, and their distance – length was estimated. Points $P1(x_1, x_2)$ and $P2(x_2, y_2)$ represent the coordinates in 2D space. Having as granted the points $(P1, P2)$ through equation$(\mathbf{19})$, we are able to calculate object's length:

$$Length = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \quad (\mathbf{19})$$

Knowing the actual length the relative error was estimated. Error estimation (Equation $(\mathbf{20})$) helped to correct system malfunctions, such as the distance between two cameras (baseline) or camera parameters.

$$Error = \frac{Actual\ Length - Measured\ Length}{Actual\ Length} \cdot 100 \quad (\mathbf{20})$$

A similar approach was used in an aquatic environment, measuring certain distances on a chessboard (Fig 37), in different depths. This was repeated several times and the relevant results are shown in Fig 38.

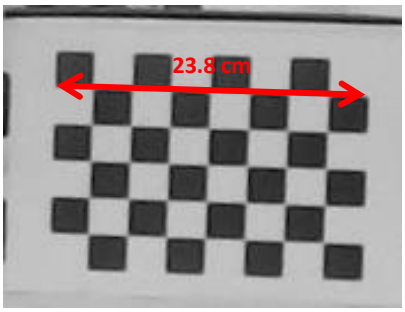


Fig 37. Known object

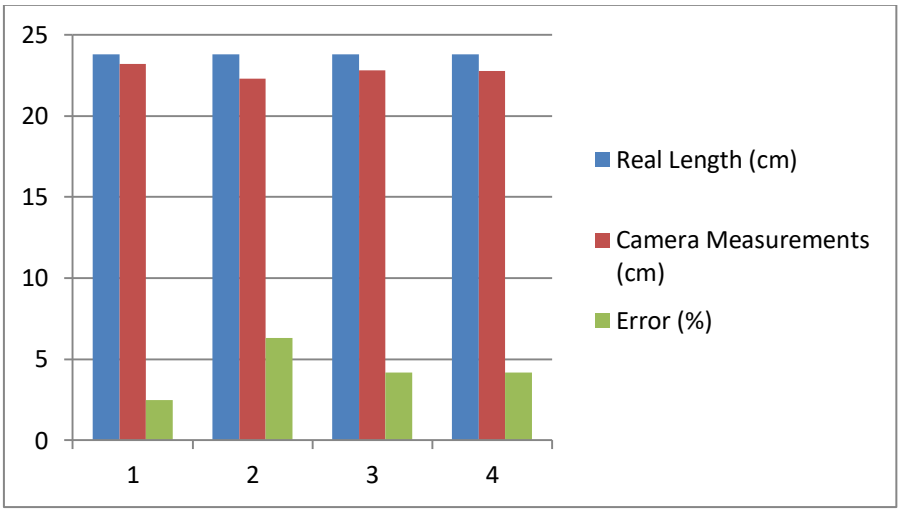


Fig 38. Measurements inside aquatic environment

3.3 Actual Fish Measurements

A final evaluation of the system took place by measuring a sufficient number of fish specimens that were placed in a tank.

3.3.1 Preparation of fish specimens

Gilthead sea breams of known sizes were placed in a fish tank (250cm x 150cm). Since the distribution range was from 20cm to 33cm, we could verify system's measurements at different distances. To identify individual specimens, a special visual color tag was placed for identification (Fig 39) following a length measurement.

Fig 39. Fish specimen with color tag

The actual measurements taken were the Total Length (TL) the Fork Length (FL) (defined as in Fig 40) and the Weight of the individuals.

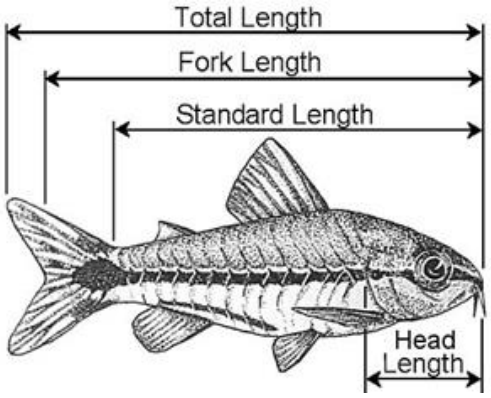


Fig 40. Definition of TL, FL, and SL parameters

3.3.2 Dataset

In order to evaluate our system a large amount of data was collected after daily monitoring of a group of 22 fish. From 100 frames collected, a two-step approach was followed. Initially, the appropriate and the inappropriate fish were selected manually. Then the previously presented PSO algorithm was applied and the selection of the specimens in each frame was compared with the ones manually selected in order to estimate the accuracy of the system.

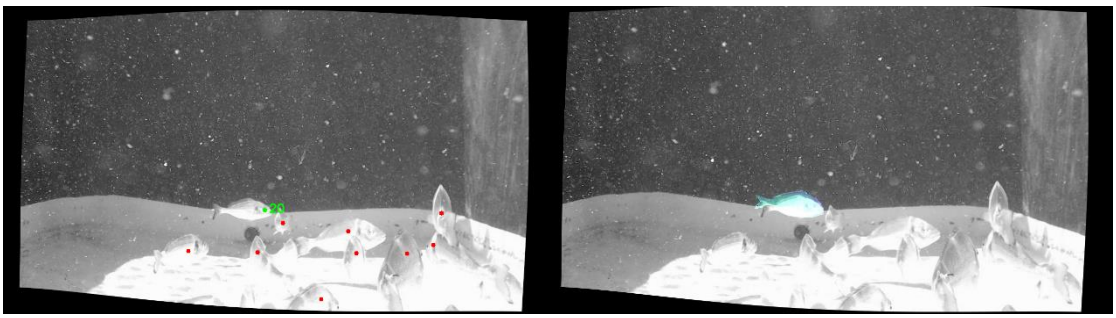


Fig 41. (a) Manually selected Appropriate fish (green dot) - Inappropriate fish (red dot); (b) algorithm selection of specimen

An example is given in Fig 41, the green dot represents the appropriate fish with its length and the red dot represents the inappropriate fish. The inappropriate fish were selected based on occlusions and incorrect inclination of the body. The system was able to detect the appropriate specimen and compare the estimated length with the actual length.

The results were analyzed based on the score (**Ω**). As mentioned in Section 2.4, the score (**Ω**) function yields a value depending on the accuracy of the model fit to the specimen identified. It is computed through equation (**16**) that outputs a value ranging from 0 to 1, where low values represent a better fit.

3.3.3 Evaluation of results

Measurements were performed on a computer with a core Intel i7 CPU, 8 GB RAM and an NVIDIA processing unit. In our project the algorithm parameters were set:

- Generations: 400
- Particles: 800

Below, the system results are analyzed based on the score as explained in the previous paragraph. The total amount of samples is 932, where the 155 samples are appropriate and the 777 samples are inappropriate. The algorithm detected 88 appropriate samples from the 100 frames analyzed. The number of proper fish per frame depended on the score (**Ω**) value, the total number of fish in the frame and the optical noise. In order to better visualize the results, three categories were created depending on the score values

- from 0.5 to 0.6
- from 0.6 to 0.7
- from 0.7 to 0.8

The best fittings were obtained at the first category, yielding relatively low error rates. As observed in Fig 42, the higher the score value the lower the fitting of the 3D model from the candidate fish.

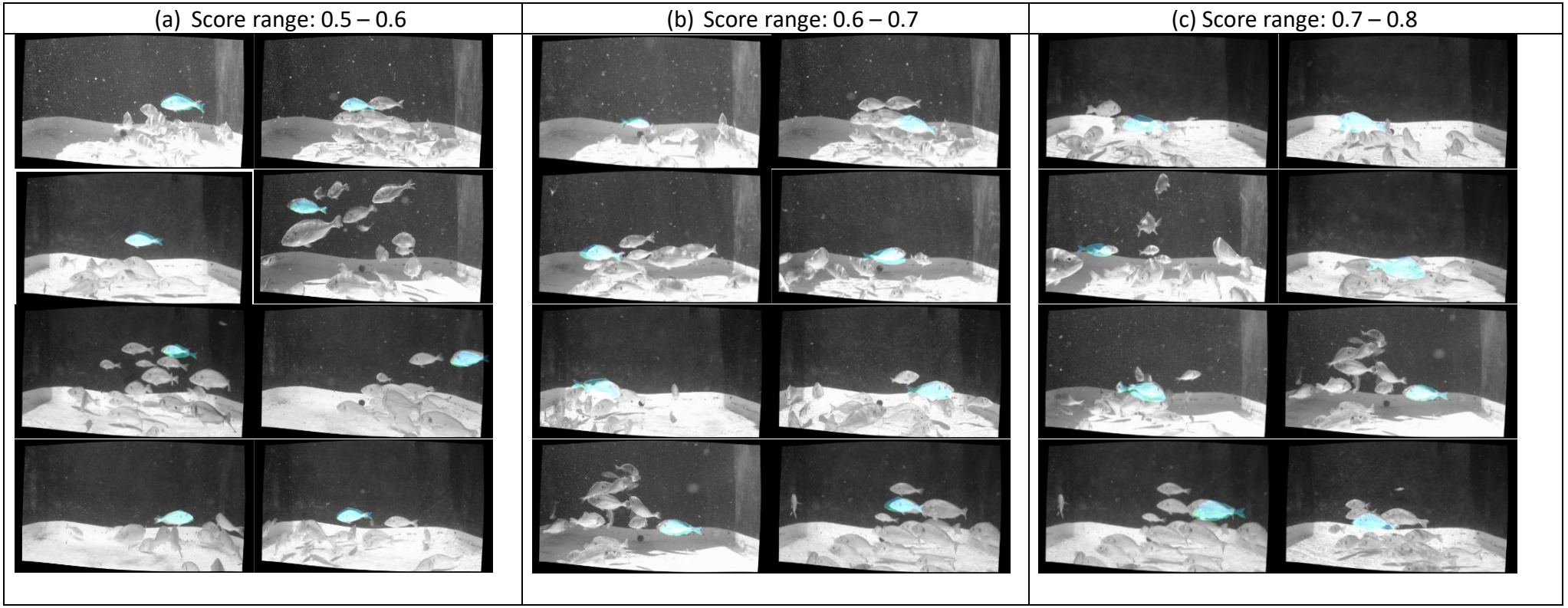


Fig 42. Indicative results in three different score ranges: (a) 0.5-0.6, (b) 0.6-0.7, and (c) 0.7-0.8

In Table 2, the results of the whole process are summarized. It shows for each score value the total number of appropriate fish identified compared to the actual specimen existed in the frames. Additionally the table shows the number of inappropriate fish identified.

Each appropriate fish identified then its total length measurement was taken and compared with the actual known value. The results of this comparison are also presented in the table as the mean error for each score category. As shown, the mean absolute error ranges from 3.3% to 16.6%. The highest errors values (16.6±20.5%; mean absolute and standard deviation) were recorded for the highest score categories of 0.75 and 0.8. On the contrary the lowest error values were observed for the 0.5 score category with a value of 3.3% while the category of 0.55-0.6 gave also acceptable error values (8.4±7.5%). For score values higher than this, although more appropriate fish are identified, the error values increase above 10%, which is considered as an unacceptable value.

Score	Appropriate Samples	Mean Absolute Error (%)	Standard Deviation (%)	Inappropriate Samples
0.5	1	3.3	-	0
0.55	7	8.4	7.5	1
0.6	12	12.4	10.7	2
0.65	27	13.9	17.7	9
0.7	36	13.9	16.7	16
0.75	49	16.6	20.6	22
0.8	50	16.6	20.4	25

Table 2. Various scores with various results of the algorithm

The most important reason for the low success rate in the identification of fish is the optical noise that affects the depth display and it can be attributed to the low transparency of the water in an aquaculture environment. An additional issue, difficult to be addressed, is the continuous movement of fish, creating occlusions and self-occlusions, making harder the proper operation of the algorithm.

Also in Fig 43 the so-called *Recall* and *Precision* diagrams are provided. The diagrams are related to an evaluation variable (score) (Table 3), which displays the optimal solution that depends on the best fit of the 3D model on the candidate fish. So the diagrams explain the precision of the PSO algorithm as the score changes. In particular it shows (a) the ratio of the identified appropriate fish to the total number of fish (Recall) and (b) is the ratio of the identified appropriate fish to the total number of appropriate fish (Precision). Analytically the formulas of Recall and Precision are:

$$Recall = \frac{t_p}{t_p + f_n} \quad , \quad Precision = \frac{t_p}{t_p + f_p}$$

where the variable t_p is referred to the appropriately identified fish, the variable f_p is referred to the inappropriately identified fish and the variable f_n is the difference of the total amount of appropriate fish minus the variable t_p ($f_n = Total\ amount\ of\ appropriate\ fish - t_p$).

Score	tp	fp	fn	Recall	Precision
0.5	1	-	154	0.0064	1
0.55	7	1	148	0.0452	0.875
0.6	12	2	143	0.0774	0.8571
0.65	27	9	128	0.1742	0.75
0.7	36	16	119	0.2322	0.6923
0.75	49	22	106	0.3161	0.6901
0.8	50	25	105	0.3226	0.6667

Table 3. Recall and Precision related to score

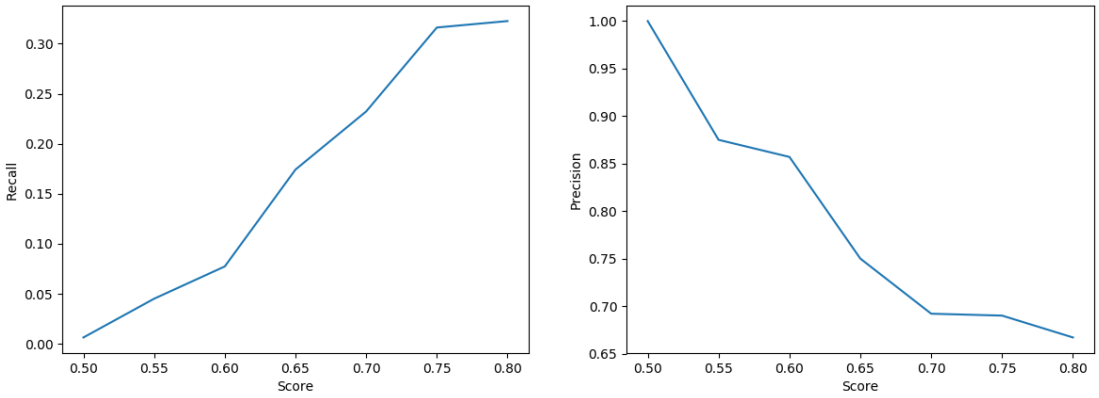


Fig 43. (a) Recall and (b) Precision diagrams

Chapter 4: Conclusions

This work was a first approach to study fish measurement with stereoscopic vision in a controlled environment and over a range of sizes. A pair of high definition cameras, managed by a mini computer, is employed to acquire a sequence of stereo frames to extract the depth map. Afterwards, a specially designed algorithm detects the appropriate candidate fish samples and through Particle Swarm Optimization algorithm, a three dimensional fish model is fitted over the shape of the fish, to estimate its length. Experiments were conducted in a controlled environment (fish tank) with 22 specimens of Gilthead Sea Bream (*Sparus aurata*), selected as a model species due to its prevalence and economic significance in the Aquaculture industry.

In order to evaluate our system, a dataset of 100 frames were selected, separating the appropriate and the inappropriate samples manually and finally we measured the tagged fish detecting from the unique color tag that each fish had. Analysis of the results indicated that the mean absolute error and the standard deviation error increase when the score value increase. The lowest error values were observed for the 0.5 score category with a value of 3.3%, on the contrary the highest mean absolute error values were observed for the 0.8 score category, where it reaches 16.6%±20.4%. The tolerance of the method to noisy observations was also evaluated. Errors in the system arise due to lesser extent of the fish tank size, further “optical” noise was added into the frame (algae, fish waste, remaining food) making it more difficult to operate 3D fitting, due to inappropriate depth mapping. Also, the fish movement was one of the factors that caused problems due to self-occlusions.

Additional research of fish measurement with stereoscopic vision is a subject for further refinements creating optimum “conditions” to the system, increasing accuracy with small error rates. Our results indicate that the proposed method is quite promising for further research, having even commercial prospects, since it represents an automated, non-invasive technology for fish size estimation, which could potentially replace the manual practices currently in use by the Aquaculture industry that are associated with high labor costs and considerable livestock stress.

Chapter 5: References

1. R. Tillett, N. McFarlane, and J. Lines, “Estimating dimensions of free-swimming fish using 3-d point distribution models,” *Comput. Vis. Image Understanding*, vol. 79, pp. 123–141, 2000.
2. L. Sola, A. Moretti, D. Crosetti, N. Karaiskou, A. Magoulas, A.R. Rossi, M. Rye, A. Triantafyllidis, and C.S. Tsigenopoulos, “*Gilthead seabream - Sparus aurata*”
3. M. Sfakiotakis, D. M. Lane, and J. Bruce C. Davies, “*Review of Fish Swimming Modes for Aquatic Locomotion*”, IEEE JOURNAL OF OCEANIC ENGINEERING, VOL. 24, NO. 2, APRIL 1999
4. J.Kennedy; R. C. Eberhart, *Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia, 1995; IEEE Service Center: Piscataway,NJ, 1995; Vol. IV, pp 1942-1948.
5. K. Thiemo*, Swarm Intelligence*, EVALife Group, Dept. of Computer Science, University of Aarhus
6. D. J.White, C. Svellingen, and N. J. C. Strachan. "Automated measurement of species and length of fish by computer vision." *Fisheries Research* 80.2-3 (2006): 203-210
7. M. Rochet, , J. Cadiou, and V. M. Trenkel. "Precision and accuracy of fish length measurements obtained with two visual underwater methods." *Fishery Bulletin* 104.1 (2006): 1-9.
8. N. J. C. Strachan, "Length measurement of fish by computer vision." *Computers and electronics in agriculture* 8.2 (1993): 93-104.
9. M. R. Shortis, et al. "A review of techniques for the identification and measurement of fish in underwater stereo-video image sequences." Videometrics, Range Imaging, and Applications XII; and Automated Visual Inspection. Vol. 8791. International Society for Optics and Photonics, 2013.
10. J. R. Martinez-de Dios, C. Serna, and A. Ollero. "Computer vision and robotics techniques in fish farms." Robotica 21.3 (2003): 233-243.
11. E. Harvey, et al. "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 63.3 (2003): 315-326.
12. R. Tillett, M. Nigel, and J. Lines. "Estimating dimensions of free-swimming fish using 3D point distribution models." Computer Vision and Image Understanding 79.1 (2000): 123-141.
13. E. S. Harvey, and M. R. Shortis. "Calibration stability of an underwater stereo-video system: implications for measurement accuracy and precision." Marine Technology Society Journal 32.2 (1998): 3-17.
14. J. Lavest, G. Rives, and J. Lapresté. "Dry camera calibration for underwater applications." Machine Vision and Applications 13.5-6 (2003): 245-253.
15. M. R Shortis., and E. S. Harvey. "Design and calibration of an underwater stereo-video system for the monitoring of marine fauna populations." International Archives of Photogrammetry and Remote Sensing 32 (1998): 792-799.
16. I. Oikonomidis, N. Kyriazis, and A. A. Argyros. "Tracking the articulated motion of two strongly interacting hands." Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012.
17. I. Oikonomidis, N. Kyriazis, and A. A. Argyros. "Efficient model-based 3d tracking of hand articulations using kinect." BmVC. Vol. 1. No. 2. 2011.
18. I. Oikonomidis, N. Kyriazis, and A. A. Argyros. "Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints." (2011): 2088-2095.

19. L. Mussi, Luca, I. Spela, and S. Cagnoni. "Markerless articulated human body tracking from multi-view video with GPU-PSO." International Conference on Evolvable Systems. Springer, Berlin, Heidelberg, 2010.
20. X. Zhang, W. Hu, X. Wang, Y. Kong, N. Xie, H. Wang, ... & Maybank, S. (2010, June). A swarm intelligence based searching strategy for articulated 3D human body tracking. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on (pp. 45-50). IEEE.
21. M. K. Alsmadi, et al. "Fish recognition based on robust features extraction from size and shape measurements using neural network." Journal of Computer Science 6.10 (2010): 1088.
22. U. A. Badawi, M. Alsmadi, and M. Khalil. "A hybrid memetic algorithm (Genetic Algorithm and Great Deluge Local Search) with back-propagation classifier for fish recognition." International Journal of Computer Science Issues (IJCSI) 10.2 Part 1 (2013): 348.
23. M. Alsmadi, et al. "Fish recognition based on the combination between robust feature selection, image segmentation and geometrical parameter techniques using Artificial Neural Network and Decision Tree." arXiv preprint arXiv:0912.0986 (2009).
24. P. X. Huang, B. J. Boom, and R. B. Fisher. "Underwater live fish recognition using a balance-guaranteed optimized tree." Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.
25. D. B. Reeder, J. M. Jech, and T. K. Stanton. "Broadband acoustic backscatter and high-resolution morphology of fish: Measurement and modeling." The Journal of the Acoustical Society of America 116.2 (2004): 747-761.
26. S. Butail, and D. A Paley. “Three-Dimensional Reconstruction of the Fast-Start Swimming Kinematics of Densely Schooling Fish.” Journal of the Royal Society, Interface 9, no. 66 (January 2012): 77–88. https://doi.org/10.1098/rsif.2011.0113.
27. Z. Chen, J. Shen, T. Fan, Z. Sun, and L. Xu. “Single-Camera Three-Dimensional Tracking of Underwater Objects.” International Journal of Signal Processing 8, no. 2 (2015): 89–104. https://doi.org/10.14257/ijsip.2015.8.2.10.
28. J. N. Fabic, I.E. Turla, J.a. Capacillo, L.T. David, and P. C. Naval. “Fish Population Estimation and Species Classification from Underwater Video Sequences Using Blob Counting and Shape Analysis.” In 2013 IEEE International Underwater Technology Symposium (UT), 1–6. IEEE, 2013. https://doi.org/10.1109/UT.2013.6519876.
29. Z. Jun, and C.M. Clark. “Autonomous Fish Tracking by ROV Using Monocular Camera.” In The 3rd Canadian Conference on Computer and Robot Vision (CRV’06), 68–68. IEEE, n.d. https://doi.org/10.1109/CRV.2006.16.
30. M. Chuang, J. Hwang, K. Williams, and R. Towler. “Multiple Fish Tracking via Viterbi Data Association for Low-Frame-Rate Underwater Camera Systems.” In 2013 IEEE International Symposium on Circuits and Systems (ISCAS2013), 2400–2403. IEEE, 2013. https://doi.org/10.1109/ISCAS.2013.6572362.
31. “Tracking Live Fish From Low-Contrast and Low-Frame-Rate Stereo Videos.” IEEE Transactions on Circuits and Systems for Video Technology 25, no. 1 (January 2015): 167–179. https://doi.org/10.1109/TCSVT.2014.2357093.
32. E.F. Morais, M.F.M. Campos, F.L.C. Padua, and R.L. Carceroni. “Particle Filter-Based Predictive Tracking for Robust Fish Counting.” In XVIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI’05), 367–374. IEEE, 2005. https://doi.org/10.1109/SIBGRAPI.2005.36.
33. K. Müller, J. Schlemper, L. Kuhnert, and K. Kuhnert. “Calibration and 3D Ground Truth Data Generation with Orthogonal Camera-Setup and Refraction Compensation for Aquaria in Real-Time.” In International Conference on Computer Vision Theory and Applications (VISAPP), 2014.

34. T.H. Pinkiewicz, G.J. Purser, and R.N. Williams. “A Computer Vision System to Analyse the Swimming Behaviour of Farmed Fish in Commercial Aquaculture Facilities: A Case Study Using Cage-Held Atlantic Salmon.” Aquacultural Engineering 45, no. 1 (July 2011): 20–27. https://doi.org/10.1016/J.AQUAENG.2011.05.002.
35. Z. Qian, X. Cheng, and Y. Chen. “Automatically Detect and Track Multiple Fish Swimming in Shallow Water with Frequent Occlusion.” Edited by Z. Daniel Deng. PLoS ONE 9, no. 9 (September 2014): e106506. https://doi.org/10.1371/journal.pone.0106506.
36. Z. Qian, S. Cheng, and Y. Chen. “An Effective and Robust Method for Tracking Multiple Fish in Video Image Based on Fish Head Detection.” BMC Bioinformatics 17, no. 1 (December 2016): 251. https://doi.org/10.1186/s12859-016-1138-y.
37. Á. Rodríguez, M. Bermúdez, J. R. Rabuñal, and J. Puertas. “Fish Tracking in Vertical Slot Fishways Using Computer Vision Techniques.” Journal of Hydroinformatics 17, no. 2 (March 2015): 275–292. https://doi.org/10.2166/hydro.2014.034.
38. M. M. Saberioon, and P. Cisar. “Automated Multiple Fish Tracking in Three-Dimension Using a Structured Light Sensor.” Computers and Electronics in Agriculture 121 (February 2016): 215–221. https://doi.org/10.1016/J.COMPAG.2015.12.014.
39. B. Sadoul, P. E. Mengues, N.C. Friggens, P. Prunet, and V. Colson. “A New Method for Measuring Group Behaviours of Fish Shoals from Recorded Videos Taken in near Aquaculture Conditions.” Aquaculture 430 (June 2014): 179–187. https://doi.org/10.1016/J.AQUACULTURE.2014.04.008.
40. C. Spampinato, S. Palazzo, D. Giordano, I. Kavasidis, Fang-Pang Lin, and Yun-Te Lin. “COVARIANCE BASED FISH TRACKING IN REAL-LIFE UNDERWATER ENVIRONMENT,” n.d. http://fish4knowledge.eu.
41. C. J. Voesenek, R. P. M. Pieters, and J. L. van Leeuwen. “Automated Reconstruction of Three-Dimensional Fish Motion, Forces, and Torques.” Edited by Iman Borazjani. PLOS ONE 11, no. 1 (January 2016): e0146682. https://doi.org/10.1371/journal.pone.0146682.
42. S. Wang, J. Zhao, and Y. Chen. “Robust Tracking of Fish Schools Using CNN for Head Identification.” Multimedia Tools and Applications 76, no. 22 (November 2017): 23679–23697. https://doi.org/10.1007/s11042-016-4045-3.
43. M. Clerc and J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation, 6(1):58–73, 2002.
44. https://www.blender.org/
45. http://www.soyohio.org/council/soy-based-enzomeal-revolutionizes-aquaculture-feedstocks/
46. https://www.bluerobotics.com/store/watertight-enclosures/wte6-asm-r1/#configuration
47. https://support.logitech.com/en_us/product/hd-pro-webcam-c920/specs
48. https://www.hardkernel.com/main/products/prdt_info.php
49. Computer vision a modern approach. David A.Forsyth
50. https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html
51. https://bersoft.com/bimagem/help/lens_calibration.htm
52. https://opencv.org
53. https://pentairaes.com/learn-about-aquaculture/what-is-fish-sampling
54. https://en.wikipedia.org/wiki/Artificial_intelligence
55. https://www.cse.unr.edu/~bebis/CS791E/Notes/CameraParameters.pdf
56. http://repository.library.teimes.gr/xmlui/bitstream/handle/123456789/1470/IXTHAL_0587.pdf?sequence=1
57. https://fgm.com.gr/uploads/file/FGM17_WEB_ENG.PDF

58. https://en.wikipedia.org/wiki/Gilt-head_bream
59. https://aquaexcel2020.eu/sites/default/files/inline-files/AquaExcel2020_NL5_final_web.pdf_0.pdf
60. http://www.vidsync.org/HomePage
61. https://blog.sicara.com/getting-started-genetic-algorithms-python-tutorial-81ffa1dd72f9
62. https://web.eecs.umich.edu/~jjcorso/t/598F14/files/lecture_0917_calibration.pdf
63. https://en.wikipedia.org/wiki/Epipolar_geometry
64. http://www.swarmintelligence.org
65. https://pentairaes.com/vaki-biomass-daily.html