Universitat Politècnica de Catalunya
Department of Signal Theory and Communications

— PhD Thesis Dissertation —

Do-it-yourself instruments and data processing methods for developing marine citizen observatories

Author: Sergi Pons Freixes

Supervisor: Jaume Piera Fernández
Co-Supervisor: Luigi Ceccaroni
Tutor: Gabriel Montoro López

Barcelona, Spain

October, 2015

Resum

L'aigua és el recurs més important per la vida al planeta Terra, cobrint més del 70% de la seva superfície. Els oceans representen més del 70% de tota l'aigua del planeta, i és on estan concentrats més del 99.5% dels éssers vius. Un gran nombre d'ecosistemes depenen de la salut d'aquests oceans; el seu estudi i protecció són necessaris.

Grans conjunts de dades durant llargs períodes de temps i al llarg d'amples àrees geogràfiques poden ser necessaris per avaluar la salut dels ecosistemes aquàtics. El finançament necessari per aquesta recol·lecció de dades és considerable però limitat, i per tant és important trobar noves formes més rendibles d'obtenir i processar dades mediambientals marines.

La solució factible actualment és la de desenvolupar infraestructures observacionals que puguin incrementar significativament les capacitats de mostreig convencionals. En aquest estudi promovem que es pot assolir aquesta solució amb la implementació d'*Observatoris Ciutadans*, basats en la participació de voluntaris.

Els observatoris ciutadans són plataformes que integren les últimes tecnologies de la informació amb ciutadans digitalment connectats, millorant les capacitats d'observació, per desenvolupar un nou tipus de recerca coneguda com a *Ciència Ciutadana*. La ciència ciutadana té el potencial d'incrementar el coneixement del medi ambient, i dels ecosistemes aquàtics en particular, mitjançant l'ús de persones sense coneixement científic especific per recollir i analitzar grans conjunts de dades.

Creiem que les eines basades en ciència ciutadana —programari lliure juntament amb maquinari de baix cost i del tipus "fes-ho tu mateix" (*do-it-yourself* en anglès)— poden ajudar a apropar la ciència del camp oceanogràfic als ciutadans. A mesura que el gran públic participa activament en l'anàlisi de dades, la recerca esdevé també una nova via d'educació pública.

Aquest és l'objectiu d'aquesta tesis, demostrar com el programari lliure i el maquinari de baix cost "fes-ho tu mateix" s'apliquen de forma efectiva a la recerca oceanogràfica i com pot desenvolupar-se cap a ciència ciutadana.

Analitzem quatre escenaris diferents on es demostra aquesta idea: un exemple d'ús de programari lliure per anàlisi de vídeos de monitoratge de llagostes; una demostració utilitzant tècniques similars de processat de vídeo en un dispositiu in-situ de baix cost "fes-ho tu mateix" per monitoratge de fauna submarina; un estudi utilitzant programari lliure d'aprenentatge automàtic (*machine learning* en anglès) com a mètode per millorar observacions biològiques; i finalment uns resultats preliminars, com a prova de la seva viabilitat, de com un mostreig manual de mostres d'aigua podria ser reemplaçat per maquinari de baix cost "fes-ho tu mateix" amb sensors òptics.

Summary

Water is the most important resource for living on planet Earth, covering more than 70% of its surface. The oceans represent more than 97% of the planet total water and they are where more than the 99.5% of the living beings are concentrated. A great number of ecosystems depend on the health of these oceans; their study and protection are necessary.

Large datasets over long periods of time and over wide geographical areas can be required to assess the health of aquatic ecosystems. The funding needed for data collection is considerable and limited, so it is important to look at new cost-effective ways of obtaining and processing marine environmental data.

The feasible solution at present is to develop observational infrastructures that may increase significantly the conventional sampling capabilities. In this study we promote to achieve this solution with the implementation of *Citizen Observatories*, based on volunteer participation.

Citizen observatories are platforms that integrate the latest information technologies to digitally connect citizens, improving observation skills for developing a new type of research known as *Citizen Science*. Citizen science has the potential to increase the knowledge of the environment, and aquatic ecosystems in particular, through the use of people with no specific scientific training to collect and analyze large data sets.

We believe that citizen science based tools —open source software coupled with low-cost do-it-yourself hardware— can help to close the gap between science and citizens in the oceanographic field. As the public is actively engaged in the analysis of data, the research also provides a strong avenue for public education.

This is the objective of this thesis, to demonstrate how open source software and low-cost do-it-yourself hardware are effectively applied to oceanographic research and how can it develop into citizen science. We analyze four different scenarios where this idea is demonstrated: an example of using open source software for video analysis where lobsters were monitored; a demonstration of using similar video processing techniques on in-situ low-cost

do-it-yourself hardware for submarine fauna monitoring; a study using open source machine learning software as a method to improve biological observations; and last but not least, some preliminar results, as proof of concept, of how manual water sampling could be replaced by low-cost do-it-yourself hardware with optical sensors.

Agraïments

Esta tesis ha sido parcialmente financiada por el antiguo Ministerio de Educación y Ciencia, mediante los programas Junta de Ampliación de Estudios (JAE Predoc, I3P-BPD2005) y Estancias breves I3P del Consejo Superior de Investigaciones Científicas (CSIC).

També vull agrair a en Maxi de l'Institut de Ciències de Mar (ICM) i la Margarita de l'Institut de Recerca i Tecnològica Agroalimentària (IRTA) per recol·lectar les dades biològiques utilitzades en el capítol 5. Menció especial també a l'Elisa de l'ICM, per ajudar-me a entendre millor tot aquest procés de monitoratge al Delta de l'Ebre. También quiero dar las gracias a Jacopo del ICM, por darnos acceso a los vídeos de los capítulos 3 y 4, y descubrirme los estudios que había realizado antes con ellos. No quiero dejar sin mencionar a los evaluadores de la comisión de tesis, Carine del ICM y Bernat del CREAF, que habéis aceptado evaluar esta tesis en un tiempo récord.

I ara miro enrere, i recordo els 4 anys que vaig passar a la Unitat de Tecnologia Marina (UTM). Allà vaig compartir despatx amb l'Elena —t'he de dir que admiro la teva capacitat de saber centrar-te en el que realment és important per tirar endavant una tesi; ja hauria acabat la meva fa uns anys si hagués seguit el teu exemple!— i l'Isma —com trobo a faltar les nostres xerrades a última hora de la tarda al despatx. Si haguessin existit els crowdfundings llavors, qui diu que no hauríem donat llum verda a alguna de les nostres idees de negoci (que en aquell moment semblaven bones)—. En Rubén no solia ser-hi en el dia a dia, però també es mereix la seva menció corresponent —tranquil, que no he arribat a explicar a ningú el que va passar en aquell congrés...—. I si parlo dels anys a la UTM, també van ser grans companys la Núria —ja has deixat de ser una Padawan per ser una Mestre Jedi de tot dret— i en Marc —no canviïs mai—. Ei, i no em descuido de la Mireia —i el suport moral mutu durant l'estada que vam coincidir al MBARI—. I en general, tota la gent de la UTM i l'ICM amb la que he passat bones estones, sigui treballant o anant a buscar un entrepà al Beltran per anar a dinar a peu de platja.

And then I moved to the European Space Agency (ESA) in Italy, where I

left my thesis a bit (a bit too much) in stand-by. I gained experience about how to observe the sea from space, a whole new world, thanks to Simon. And there I met many people from many places: Xavi —Visca la terra!—, Pascal, Marcello, Philippe, Chandra, Alex... good parties, good food, good times. Y Clara, fué un placer descubrir Roma (particularmente sus heladerías) contigo. Y las clases extratescolares de Castellano!

And then I moved to ESA in the Netherlands, where I slowly retook the thesis while having a new full time job. And here I learned a lot about the agency, and I wish my best to all the Sentinel 3 team. Hoping to see the baby in orbit soon! E non posso parlare della Olanda senza parlare dalla mia famiglia mentre sono stato lì. Cristian, Sonia, Giulia, Flavio, Dori, Clara (tú otra vez?), Roberto... Siete stati il sole caldo che non c'era in Olanda. Mi mancate ragazzi!

También le quiero agradecer a mi co-director, Luigi, la valentía de aceptar co-dirigir esta tesis y tener paciencia conmigo. En estos años de investigación has sido mi referencia de como hacer ciencia de forma rigurosa, donde la excelencia y la profesionalidad son un factor constante. Grazie mille.

I com no, el meu director, en Jaume. El que em va engrescar —o hauria de dir liar?— a començar el doctorat i provar això que en diuen fer recerca. Sempre has avisat que no és un món on fer-se d'or, però sempre t'has preocupat per donar-me suport d'una forma o una altra quan realment no hi tenies cap obligació, i això t'ho agraeixo molt. I després de tants anys veig que segueixes tenint noves idees, una rere l'altra i totes amb passió, i per això em trec el barret. Molts acaben el doctorat dient que el seu jefe "és un cabrón". Jo, en canvi, estic molt content que hagis estat tu.

I ja ens apropem al final. I possiblement la persona que m'ha animat més a donar una empenta final a la tesi ha sigut la Laura. Realment m'has d'estimar molt per aguantar-me i animar-me com ho fas; i et mereixes una medalla per aconseguir que m'assegués davant de l'ordinador a treballar com tocava, que si no encara estaria fent proves amb dades. Gràcies a tu estic tancant una etapa de la meva vida al mateix temps que n'estic començant una de nova. I aquesta nova etapa és al teu costat, no puc demanar res més per ser feliç.

I per acabar, els meus pares, en Marcial i la Roser. Si soc on soc, és gracies a vosaltres. No només per l'educació que m'heu donat, si no per tot l'amor que sempre he sentit que se'm donava i tots els sacrificis que heu fet (i encara feu) pels vostres fills. Moltes gràcies. Us estimo.

Sergi

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The need to monitor and manage aquatic ecosystems

Water is the most important resource for living on planet Earth, covering more than 70% of its surface. The oceans represent more than 97% of the planet total water and they are where more than the 99.5% of the living beings are concentrated (Gleick et al., 1993; Mockler, 1995). A great number of ecosystems depend on the health of these oceans; their study and protection are necessary. In this line, on the political front, the United Nations General Assembly, in December 2003, proclaimed the years 2005 to 2015 as the International Decade for Action 'Water for Life' (Nations, 2003), to protect and manage water.

Protecting and managing aquatic ecosystems is a challenging task. These environments are characterized by an extraordinary mix of human activities: tourism, fishing and industry (e.g., petrochemical plants and aquaculture). Given the frequently conflicting interests between conservation and exploitation, the fate of aquatic ecosystems is often a hot political issue. The attitudes and values of stakeholders in environmental issues are an essential part of the stewardship of conflicting environments. New policies concerning environmental resources should have citizens' support and consider public attitudes from the beginning.

The development of policy is becoming more complex with larger datasets required to support the assessment of impacts on whole ecosystems over long periods of time. For example, many years of data collected over wide geographical areas can be required to assess the impact of different pressures (e.g., tourism, litter or fishing) on ecosystem health for the Marine Strategy Framework Directive (Olenin et al., 2010). The funding needed for data collection is considerable and limited, so it is important to look at new cost-effective ways of obtaining and processing marine environmental data.

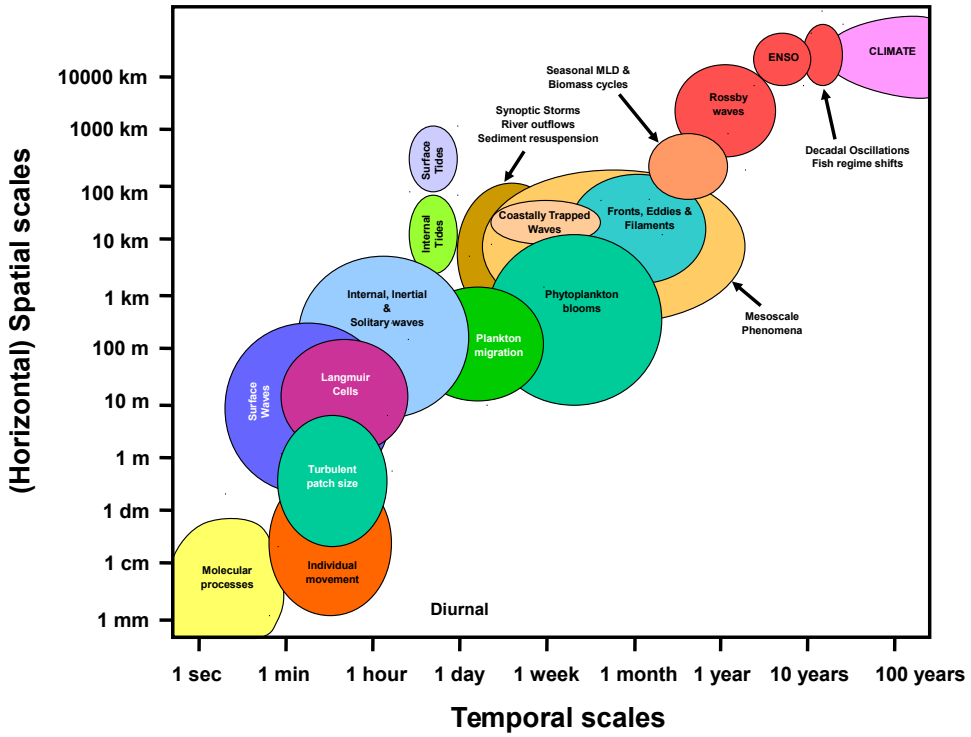


Figure 1.1: Time and space scales comparison of several natural ocean processes. Adapted from Dickey and Bidigare (2005).

1.2 Observing ocean processes: sampling strategies

During the nineteenth century oceanography consisted on measurements and models from exploratory, mapping and sampling. It has led to recognize the time-dependent complexity of processes that occur within the oceans (Person et al., 2007). Earth and its oceans are not static, they have a dynamic behavior at several scales, both temporal and spatial, as can be seen on figure 1.1 (Dickey and Bidigare, 2005).

In any plan to design the strategy to monitor ocean processes, one of the first questions to address is the number of field samples to measure and the sampling frequency to obtain them. In many cases the answers to these questions are based mainly in logistic and operational restrictions (e.g., maximum number of samples that it is possible to process analytically, instrumental requirements, or cost of data transmission). However, it is important to take into account the principles of *Information Theory* in order to avoid potential artifacts derived from improper sampling designs.

At present, there is a clear limitation on the sampling capabilities (see figure 1.2).

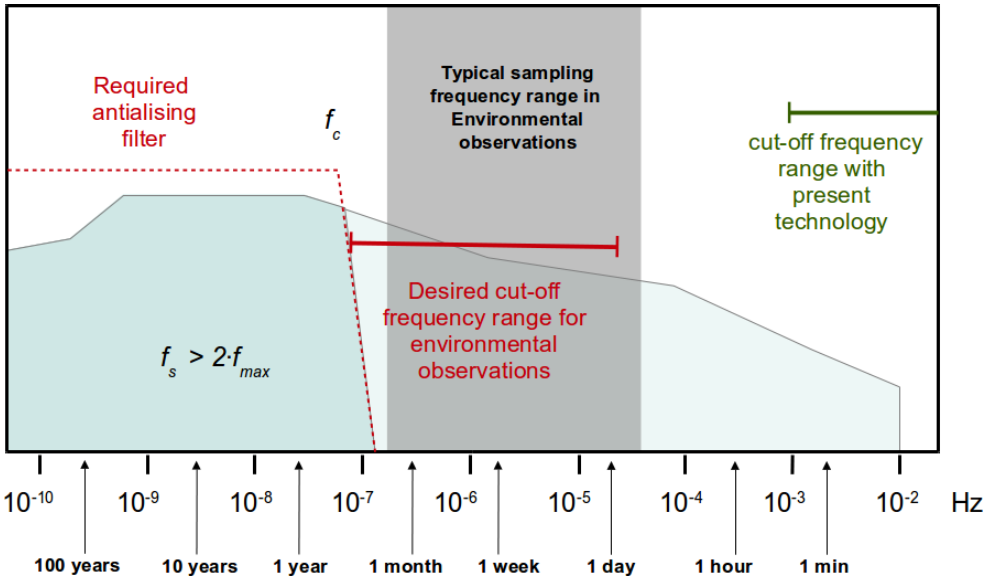


Figure 1.2: Comparison of the sampling frequencies in conventional environmental observations with the cut-off frequencies that would be needed to avoid aliasing.

According to figure 1.1 the time scales for ocean processes cover from seconds to centuries, which corresponds in the frequency domain to a frequency range of 10^{-2}–10^{-10} Hz. Sampling rates in conventional long term monitoring programs are usually monthly or weekly and exceptionally daily (i.e., sampling frequencies in the range of 10^{-7}–10^{-4} Hz), which clearly may induce aliasing artifacts according to the bandwidth of the observed processes.

To avoid aliasing, it would be necessary to implement analog anti-aliasing filters with cut-off frequencies in the range 10^{-7}–10^{-4} Hz, several orders of magnitude below the lower limit of the present technology —currently it is difficult to implement low-pass filters with cut-off frequencies below 10^{-2} Hz.

The only feasible solution at present is to develop observational infrastructures that may increase significantly the conventional sampling capabilities. At present two main solutions have been promoted:

- The implementation of complex (and costly) observatories with the most advanced technologies, particularly, those based on underwater cabled networks (figure 1.3 left). There are several examples of this type of approaches around the world, among them: Neptune from Canada (Barnes et al., 2008), Ocean Observatories Initiative (OOI) from USA (Chave et al., 2009), Monterey Accelerated Research System (MARS) from USA (Massion and Raybould, 2006), European Multidisciplinary Seafloor and water-column Observatory (EMSO) from Europe (Best et al., 2014).

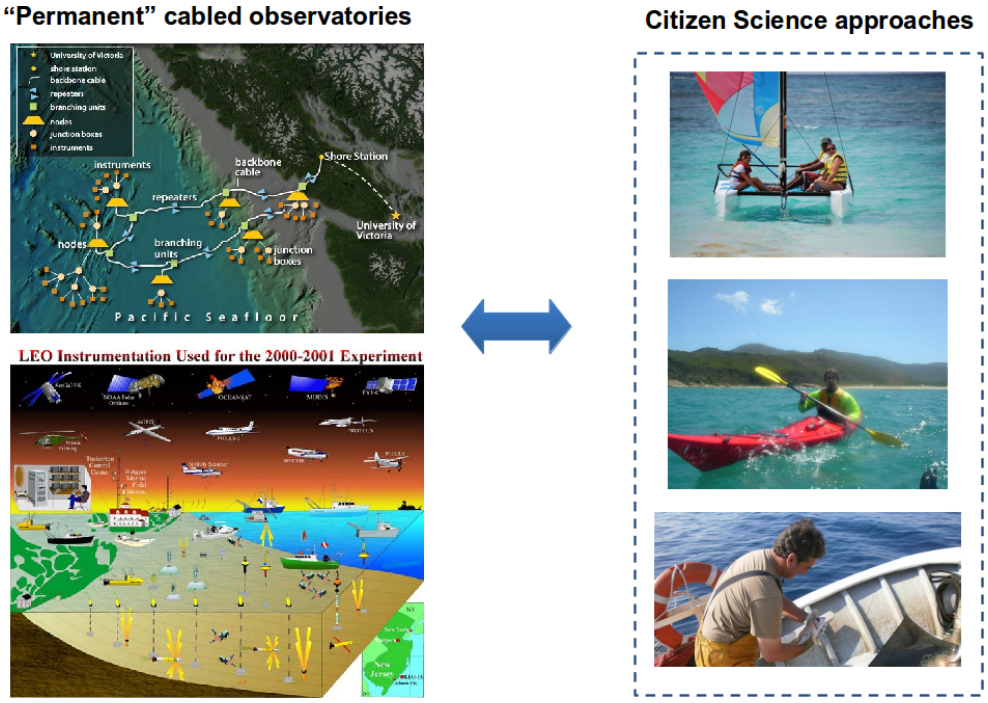


Figure 1.3: Comparison between observational infrastructures. Left: Complex underwater observatories. Right: "Citizen observatories".

- The implementation of *Citizen Observatories*, based on volunteer participation (figure 1.3 right), which will be commented in the following section.

1.3 Citizen science, enhancing observational capabilities in oceanography

Citizen observatories are a new concept of research infrastructure that is being promoted around the world and Europe in particular (Citizens' Observatory, 2015). Citizen observatories are platforms that integrate the latest information technologies to digitally connect citizens, improving observation skills for developing a new type of research known as *Citizen Science*[1].

Citizen science has the potential to increase the knowledge of the environment, and aquatic ecosystems in particular, through the use of people with no specific scientific training to collect and analyze large data sets.

[1]The term *citizen science* was defined as "scientific work undertaken by members of the general public, often in collaboration with or under the direction of professional scientists and scientific institutions" by the Oxford English Dictionary in 2014.

Until recently, citizens had a passive role in science, being generally at the end of the information chain. In reality, local communities embody a rich source of historical knowledge —commonly more extensive and/or nuanced than that held by the local authority— as well as having the potential for providing dynamic, higher resolution data (Wrigley, 2014). Crowd science projects are able to draw on the effort and knowledge inputs provided by a large and diverse base of contributors, potentially expanding the range of scientific problems that can be addressed at relatively low cost, while also increasing the speed at which they can be solved (Franzoni and Sauermann, 2014).

There is a recognized awareness of the need to meaningfully engage society in efforts to tackle marine conservation challenges (Lotze et al., 2011). The value of citizen science has been widely recognized by national governments (Pocock et al., 2014), international bodies and funding agencies (Hyder et al., 2015). A white paper has been developed on the future of European citizen science and the United Nations Environment Programme has stated that citizen science is an essential means of achieving sustainability (Au et al., 2000).

The number of projects globally that engage the public in scientific research has dramatically increased in recent years (Conrad and Hilchey, 2011). Some examples of existing citizen science programs, for aquatic ecosystems and other subjects, are:

- Secchi Dip-In (Lee et al., 1997) is a demonstration of the potential of volunteer monitors to gather environmentally important information on lakes, rivers and estuaries. The concept of the Dip-In is simple: individuals in volunteer monitoring programs take a transparency measurement on one day during the weeks surrounding Canada Day and July Fourth. Individuals may be monitoring lakes, reservoirs, estuaries, rivers, or streams. These transparency values are used to assess the transparency of the volunteer-monitored lakes rivers and estuaries in the United States and Canada.

- "USGS iCoast - Did the Coast Change?" (Liu, 2014) is a USGS research project to construct and deploy a citizen science web application that asks volunteers to compare pre- and post-storm aerial photographs and identify coastal changes using predefined tags. This crowdsourced data will help USGS improve predictive models of coastal change and educate the public about coastal vulnerability to extreme storms.

- Creek Watch (Kim et al., 2011) is an iPhone application that enables the user to help monitor the health of his local watershed. The user can take pictures of waterways using the Creek Watch application and report how much water and trash he can see. Data is aggregated and shared with

water control boards to help them track pollution and manage water resources.

- The Citclops project (Wernand et al., 2012) aims to develop systems to retrieve and use data on seawater color, transparency and fluorescence, using low-cost sensors combined with people acting as data carriers, contextual information (e.g., georeferencing) and a community-based Internet platform. Methods are being developed to rapidly capture the optical properties of seawater, e.g., color through Forel-Ule observations, and transparency through a variant of the Secchi disc. People will be able to acquire data taking photographs of the sea surface on ferries or other vessels, on the open sea or from the beach.

- The Quake-Catcher Network (Cochran et al., 2009) is a collaborative initiative for developing the world's largest, low-cost strong-motion seismic network by utilizing sensors in and attached to internet-connected computers. The Quake-Catcher Network can provide better understanding of earthquakes, give early warning to schools, emergency response systems, and others. It also provides educational software designed to help teach about earthquakes and earthquake hazards.

- Wildlife@Home (Desell et al., 2013) combines both volunteer computing, where people volunteer their computers to different computing projects, and crowdsourcing, where people volunteer their brain power, to aid in the analysis of a vast amount of video. Wildlife@Home is used to compare the results of preliminary motion and feature detection algorithms to the validated observations made by users.

- Regarding the classification of still images, GalaxyZoo (Lintott et al., 2008) has had great success in using volunteers to classify galaxies in images from the Sloan Digital Sky Survey; and PlanetHunters (Fischer et al., 2012) has been used to identify planet candidates in the NASA Kepler public release data. Snapshot Serengeti has been created to classify images from camera traps in the Serengeti National Park (Hines et al., 2015).

- Wrigley (2014) proposes a core citizens' observatory theme by deploying local citizens as "social sensors" and engaging them in citizen science. To this end, two approaches have been developed: an Android phone app and a Raspberry Pi[2] sensing device. The former allows static sensor readings (e.g., gauge board levels), qualitative reports and photographs to be submitted. The latter provides a low-cost, mobile device that records atmospheric conditions (e.g., temperature, barometric pressure

[2]More information about the Raspberry Pi is shown in chapter 4.

and luminosity) as well as a means of estimating river flow using the Raspberry Pi camera module.

- The Zooniverse (Borne and Team, 2011) is a citizen science web portal run by the Citizen Science Alliance. Volunteers contribute for free to projects from astronomy, ecology, cell biology, humanities and climate science.

Within the citizen science framework there are different schemes of collaboration:

Crowdsourcing is the process of obtaining needed services, ideas, or content by soliciting contributions from a large group of people, and especially from an online community, rather than from traditional employees or suppliers. It combines the efforts of numerous self-identified volunteers or part-time workers, where each contributor of their own initiative adds a small portion to the greater result.

Do-It-Yourself (DIY) is the concept in which we may include all kind of initiatives of open source hardware and software tools and methods to generate knowledge and share data. Although it is a very appealing collaboration approach, it is well known from existing studies of DIY technologies (Krebs, 2010) that the ability to actively participate in DIY activities requires free time, knowledge, and social capital to access or buy equipment, etc. These aspects can make these practices exclusionary on multiple levels. Some people are excluded because of confidence, others on the basis of education and yet other because of financial issues.

Do-It-Together (DIT) is a concept that has been used mainly on the development of large open source projects, but it can be extended to citizen science as well. Do-it-together activity, as its name, is the kind of activity which single user cannot accomplish. It needs many users to work together at the same time. Alternative activity needs multiple users' effort, each one's activity is connected to former ones (Gao-feng et al., 2006).

The *DIY-DIT Movement* promotes important values, such as curiosity, tinkering, collaborative problem-solving, and self-efficacy. Apart from that, it is important for a number of reasons (National Economic Council and Office of Science and Technology Policy, 2015):

1. Students are engaged easily in DIY-DIT projects, revitalizing career and technical education and inspiring students to excel in Science, Technology, Engineering, and Mathematics (STEM).

2. Adults may gain, participating in making their own DIY devices, the skills they need for jobs in fields such as design and advanced manufacturing.

3. DIY-DIT activities lower the barriers to entrepreneurship in hardware and manufactured products, in the same way that cloud computing and open source software have lowered the costs of launching an Internet-based startup.

Chapter 2

Objectives

In a citizen science based monitoring scenario, the information donated voluntarily by citizens can be used as a low-cost labor way to find solution to the sampling challenges stated in the previous section. We believe that citizen science based tools (open source software coupled with low-cost do-it-yourself hardware) can help to close the gap between science and citizens in the oceanographic field. As the public is actively engaged in the analysis of data, the research also provides a strong avenue for public education. We could be able to perform outreach to future oceanographers by providing accessible tools to volunteers as well as by engaging computer scientists in the projects' open source code development.

In this thesis several monitoring scenarios in different marine related disciplines will be analyzed to implement, in each case, solutions based on citizen science instruments (DIY) and/or data processing chains. The main goal is to demonstrate how citizen science tools —based on open source software and low-cost hardware— are effectively applied to solve the requirements for monitoring the analyzed processes.

In this thesis, four different scenarios are tackled:

- On chapter 3, video images from laboratory that were previously manually inspected are now processed using open source software. Manual analysis is substituted by automatic analysis, open to be reused and improved by other scientists or any passionate about video image processing.
- Chapter 4 addresses the analysis of video footage from an underwater camera that has been before analyzed using proprietary software on a desktop workstation. On our approach the video is processed by open source software on a small device that can be purchased on a electronics

store for just 35USD, allowing any one to have its own observational node.

- Chapter 5 involves an innovative research that has not been performed before: the use of time-series of biological measurements to predict harmful algal blooms on the Alfacs Bay. The computation is performed using open source that can be reviewed an improved by the community.
- Chapter 6 goes one step forward than the previous chapter. The laboratory analysis used as data source for the prediction is now replaced by hypothetical low-cost do-it-yourself optical sensors.

Chapter 3

Automated video-image processing for animal behaviour analysis using open source tools

This chapter is based on the previous author's work:

Automated video-image analysis for the analysis of the behaviour of deep-water lobsters (Nephrops norvegicus)
Conference talk and proceedings in *3rd MARTECH International Workshop On Marine Technology*, Vilanova i la Geltrú (Spain), 2009.

Video-image processing applied to the analysis of the behaviour of deep-water lobsters (Nephrops norvegicus)
Finalist in student poster competition in *IEEE/OEE Oceans Conference and Exhibition, Oceans'10*, Sydney (Australia), 2010.

3.1 Introduction

The objective of this chapter is to introduce open source video processing systems as a method to improve biological observations. In this case, under laboratory experimental conditions.

One aspect of animal behavior consists of movement patterns linked to physical or biological environmental variables, which may be of a cyclic nature. Rhythmic behavior, in particular, encompasses all motor acts that involve a rhythmic repetition coupled to a cyclical variable (Naylor, 1988; Nusbaum and Beenhakker, 2002). Among rhythmic behavioral processes, locomotion is the most widely studied indicator in biological clock regulation research (Naylor, 1988; Ortega-Escobar, 2002).

Activity rhythms of important fishery resources are interesting for its effect on their catchability (Naylor, 2005). The laboratory study of activity

rhythms in deep water species is to date limited by technical constrains related to the peculiar animals' environment (Menesatti et al., 2009). A widely adopted methodological solution has been to date the use of infrared (IR) actography (Aguzzi et al., 2008)). But highly corrosive salt water impairs the correct functioning of hardware equipment made by IR emitting and detecting barriers. In this context, video image analysis represents a more simple hardware solution when a proper level of automation image processing is acquired (Aguzzi et al., 2009a).

Video image analysis can be efficiently used to disclose the period of activity rhythms in relation to the day-night cycle over a long period of time in laboratory controlled conditions (Aguzzi et al., 2004). The video image analysis of footage depicting the behavioral pattern of species in relation to time is to date of growing interest for neuroethology and biomedicine (Aguzzi et al., 2009b). But when data volumes grow dramatically, manual analyses become completely impractical in many domains. Hence, analysis requires automation (Fayyad et al., 1996).

In this chapter, we show the results of using open source software to apply automated data processing to a set of videos previously used to manually monitor the period of activity rhythms of the deep-water Norway lobster *Nephrops norvegicus*, a species of elevated commercial fishery value.

3.2 Materials and Methods

3.2.1 Data description

On the original study of Aguzzi et al. (2004), 28 adult lobster males of the species *Nephrops norvegicus* were monitored. These individuals were collected by a commercial trawler in the western Mediterranean Sea a few miles off Barcelona. Each animal was individually housed in a plastic tank of $40 \times 25 \times 20$ cm, supplied with an external pump providing the appropriate circulation and filtration of water. The animals were not fed during the experiments to prevent interference because of food presence-absence, which directly affects locomotor activity and metabolism.

Infrared-sensitive video cameras were located over the tanks, pointing vertically downwards to obtain a view from the top. Each camera had three complete tanks on its field of view. A constant source of infrared light illuminated the tanks also from above. The video cameras were connected to a time-lapse VHS video tape recorder, capturing one frame every 1.7 seconds and storing them as a 25 frames per second footage.

For our study, two of the tapes (for a total of six tanks) were digitized by means of Pinnacle Studio application, producing two video files encoded with the DVSD codec, AVI container and a resolution of 720×576 pixels. A sample is shown in figure 3.1.

Figure 3.1: Sample frame from one of the VHS tapes after digitization.

3.2.2 Processing

All the video image processing described on this section was performed in C, making use of the open source library OpenCV (Bradski, 2000; Pulli et al., 2012). OpenCV is a library of programming functions for real time computer vision; initially an Intel Research initiative launched in 1999, and taken over in 2012 by the non-profit foundation OpenCV.org.

The footages were first split according to position and size of the tanks, obtaining three new video streams per original video, in order to analyze each tank separately.

Next, a Gaussian smoothing was applied to each video to reduce the noise level, at expenses of reducing the sharpness of the image. Preliminary tests showed that this trade-off improved the results of the edge-detection algorithm used posteriorly.

Before choosing a motion quantification technique, the characteristics of the video must be analyzed. Motion detection of objects moving over a scenario consists on obtaining a model of the background, with no foreground objects present. Any change on this modeled background is an indicator of presence, i.e., there is a foreground object on it. When the scenario is a static scene, e.g., an empty room under the effect of invariable light conditions, a single frame is enough to create the background model. In our case we did not have a completely static background, as we had random appearances of:

- Bubbles produced by the water refreshing circuit.

- Reflections of artificial light sources on the waves of the water surface.
- Snow noise due to the quality of the original VHS tapes, that was not removed during the smoothing preprocessing.
- Non-uniform illumination, i.e., areas of low contrast in some parts of the tanks.

Therefore we needed a more complex model than a single static frame. We implemented a technique known as codebook background subtraction (Kim et al., 2005; Bradski and Kaehler, 2008), which creates dynamic background models. Codebook background subtraction works at a pixel level. When the values of a pixel fluctuate between a defined margin for a set period of time, a codebook is created around these values. Any future pixel value inside this codebook is considered background, and the rest foreground. If the new values of the pixel are repeated along a set time instead of the ones of the codebook, the old codebook is deleted and a new one is created. Therefore, the whole set of codebooks of all the pixels represent a compressed form of background model for a long image sequence, and allows to capture structural background variation due to periodic-like motion over a long period of time under limited memory. Figure 3.2 illustrates this concept.

Once the background is modeled, each frame is transformed to a binarized image with black background and white foreground objects. In order to eliminate non interesting small objects, such as bubbles or waves, the image is subjected to a process of erosion-dilation (Serra, 1983). This process has also the effect of smoothing the contours of the objects, as exemplified on figure 3.3.

On each binarized frame, cleaned after the erosion-dilation process, all the objects are detected with a contour finding algorithm (Suzuki et al., 1985), where a contour is defined by the edge between a black zone and a white zone. The object with the biggest contour is assumed to be the lobster. The amorphous contour of the lobster is simplified to a bounding box (see figure 3.4), and the center of the box is calculated, assuming that it is approximately the center of the lobster. A list with the centers of the lobster on each frame is stored to a log file.

From this point forward, the rest of the processing was performed in Python with the use of NumPy and SciPy (Jones et al., 2001; Oliphant, 2007), ingesting the log file created on the previous step. SciPy is a Python-based ecosystem of open source software for mathematics, science, and engineering. NumPy is one of the core packages of SciPy, supporting large multidimensional arrays and matrices and providing high-level mathematical functions to operate on these arrays.

On the original manual analysis of the videos, motion was quantified by counting how many times the individual crossed from one half of the tank to the other half, in periods of 30 minutes. On our study, motion is quantified

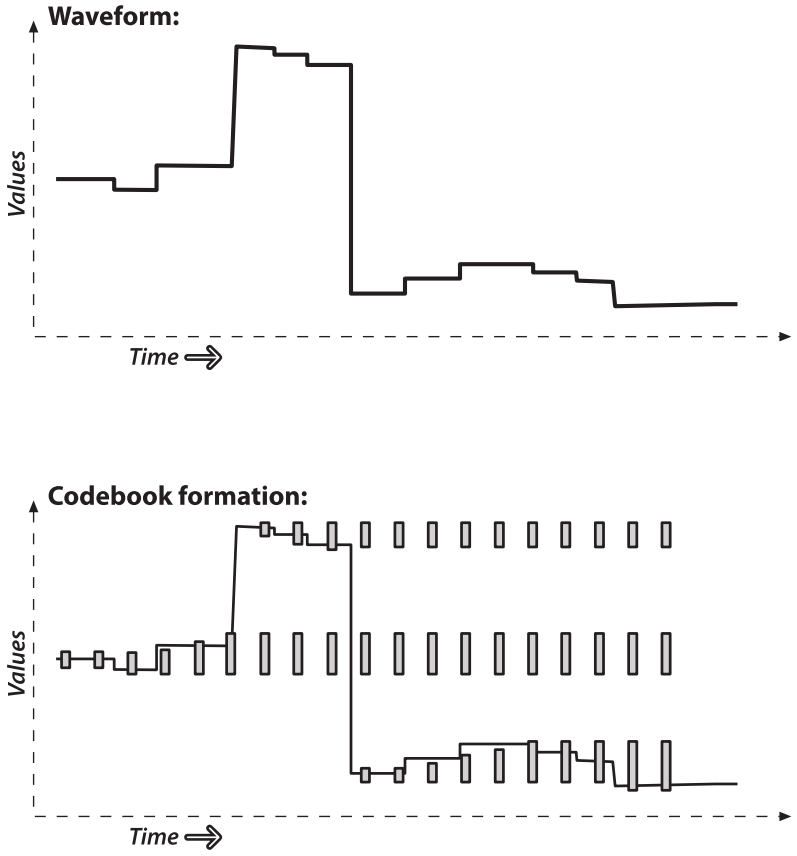


Figure 3.2: Codebooks delimit intensity values repeated over time, considered "background". A box is formed to cover a new value and slowly grows to cover nearby values; if values are too far away then a new box is formed. With permission from Gary Bradski ©2008 (Bradski and Kaehler, 2008).

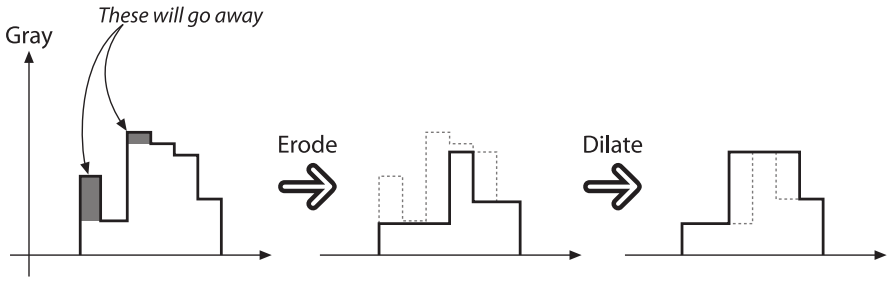


Figure 3.3: Example of erosion-dilation. The upward outliers are eliminated as a result. With permission from Gary Bradski ©2008 (Bradski and Kaehler, 2008).

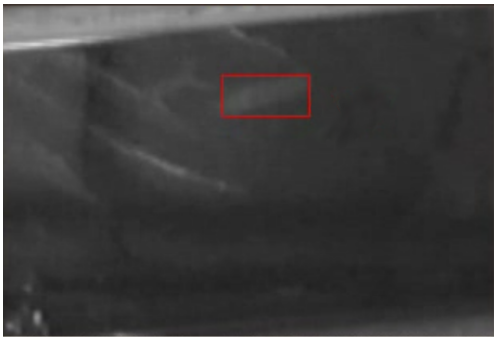


Figure 3.4: Bounding box automatically created around the main foreground object, i.e., the lobster. The center of the box is used as indicator of the specimen position

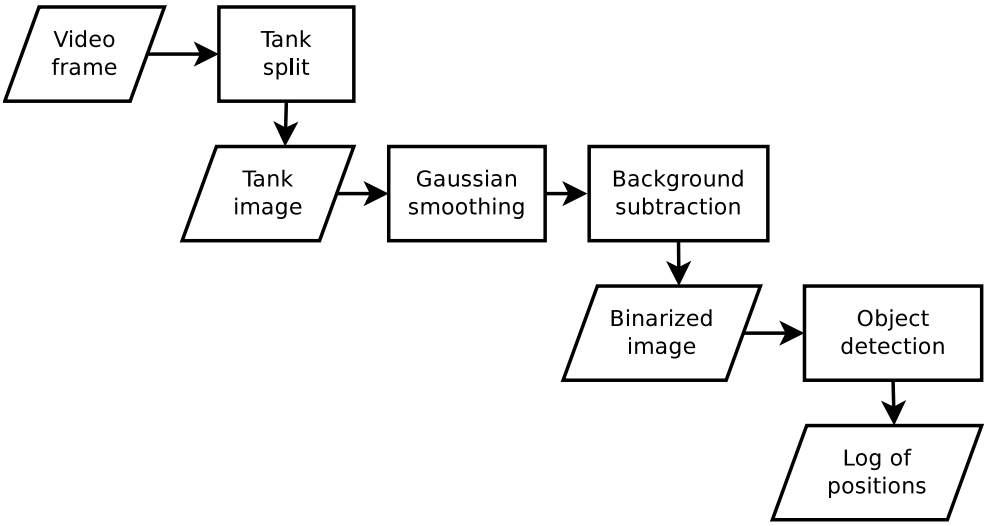


Figure 3.5: Complete video processing chain.

by measuring the euclidean distance between the centers of the lobster from frame to frame. To ease the comparison between the two methodologies, the new data was rebinned to periods of 30 minutes by calculating the sum of distances of this period.

Initially we implemented an alternative approach to quantify motion, but it was discarded in favor of the aforementioned methodology as it provided best results overall. On this alternative approach, the center of the objects was not detected and stored. Instead, the difference between two consecutive binarized images was quantified. This difference was stored to a log file, and the processed following the instructions above. Figure 3.6 shows a time series where both methodologies are compared, with also the original data. More

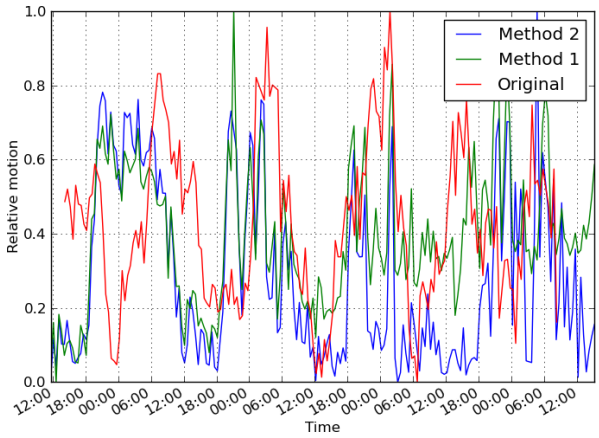


Figure 3.6: Sample of time series of relative motion quantified according to three different methodologies for the same video. Method 2 shows a more marked periodicity than the other two, particularly on the second half of the series. Original: original study. Method 1: based on the difference of binary images. Method 2: based on the difference between centers. This is the methodology chosen for our study.

information about this alternative approach can be found in appendix B.

Measurement uncertainty and noise sometimes make it difficult to spot oscillatory behavior in a signal, even if such behavior is expected. The autocorrelation[1] sequence of a periodic signal has the same cyclic characteristics as the signal itself. Thus, finding the peaks in the autocorrelation can help verify the presence of cycles and determine their durations (Rosenfeld and Troy, 1970). We calculated the autocorrelation for our time series, and then relative maxima of the autocorrelation to find the peaks.

Prior to the autocorrelation, the mean was subtracted from the signal and it was smoothed using a median filter to improve the peak detection.

The full processing chain is shown in figure 3.5.

3.3 Results

We had access to the original data of the processed videos —i.e., the time series measuring how many times the individuals crossed from one half of the tank to the other half—. The processing applied to our metrics of motion —i.e., the output of the image processing chain— was also applied to the original data for the sake of comparison. The peaks found on the autocorrelation for each

[1] Autocorrelation is the cross-correlation of a signal with itself at different points in time. In signal processing, cross-correlation is a measure of similarity of two series as a function of the lag of one relative to the other. In an autocorrelation there will always be a peak at a lag of zero, and its size will be the signal power.

	Peaks locations (hours)	
Tank ID	Original data	New data
a	23.5	10.5, 23.5
b	23.5	17.5, 22.5
c	16.5	23
d	24	19.5, 24
e	13	6.5, 13.5, 23.5
f	12, 24	6, 12.5, 19.5

Table 3.1: Local maxima (peaks) detected on the autocorrelation of the motion signal for the original manual methodology and the new automated methodology.

tank of the videos are shown on table 3.1. For each tank, the peak detected corresponds to the periodicity of the signal. A periodic signal repeats itself over time. Detection of a single peak (original data cases *a*–*e*, new data case *c*) means that the signal has single periodicity. Detection of multiple peaks (original data case *f*, new data cases *a*, *b*, *d*–*f*) means that the signal has multiple periodicities, i.e., the signal has more than one underlying periodic component. This is difficult, if not impossible, to spot by eye (Leis, 2011).

3.4 Discussion

According to the original article, the individuals were found to follow three different possible patterns:

- Circadian locomotor rhythmicity with recorded significant periods between 20 and 25 h.
- Ultradian periodicity of around 12 h.
- Ultradian periodicity of around 18 h.

Table 3.2 expresses table 3.1 assigning the peaks to the closest of the known circadian and ultradian periodicites.

Cases *a*, *b* and *d* show the same 24 hours circadian periodicity than the original study. Additionally, ultradian periodicities were found on the new data: what it seems a 12 hours periodicity for *a* and 18 hours for *b* and *d*.

Case *e* (figure 3.8) has a 12 hours ultradian periodicity on both methodologies, but also a 24 hours circadian periodicity was found on the new one. Additionally, a periodicity of 6.5 hours was detected.

Case *f* has a 12 hours ultradian periodicity on both methodologies. A clear 24 hours circadian periodicity was only found on the original data, while the

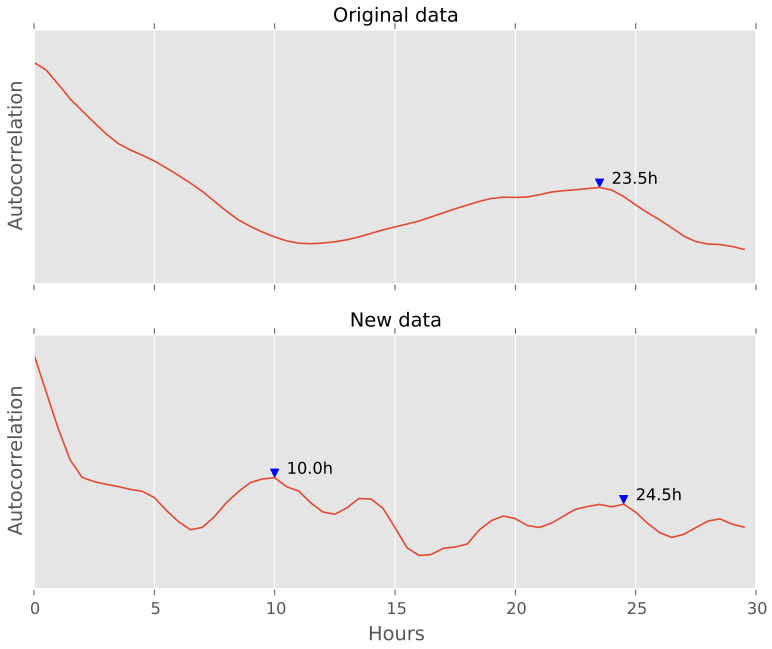


Figure 3.7: Detail of the autocorrelation for video *a*. A peak at ∼24 hours indicates a circadian periodicity on the original data. Peaks at ∼24 and ∼12 hours on the new data indicate circadian and ultradian periodicities respectively.

new data shows it at 19.5 hours. It could be considered a 18 hours ultradian case. As in case *e*, a periodicity of 6 hours was detected also.

Case *c* (figure 3.7) showed what it could be considered a 18 hours ultradian periodicity (16.5 hours real) in the original data, while it was a 24 hours circadian periodicity (23 hours real) with the new data.

To conclude:

- From the 7 periodicities on the original data, 5 of them have been also detected by the automatic methodology.
- For the other 2 peridocities of the original data, alternative periodicites were found on the new data (e.g., a 24 hours ultradian instead of a 18 hour circadian.)
- 4 more periodicities were found on the new data compared to the old data.

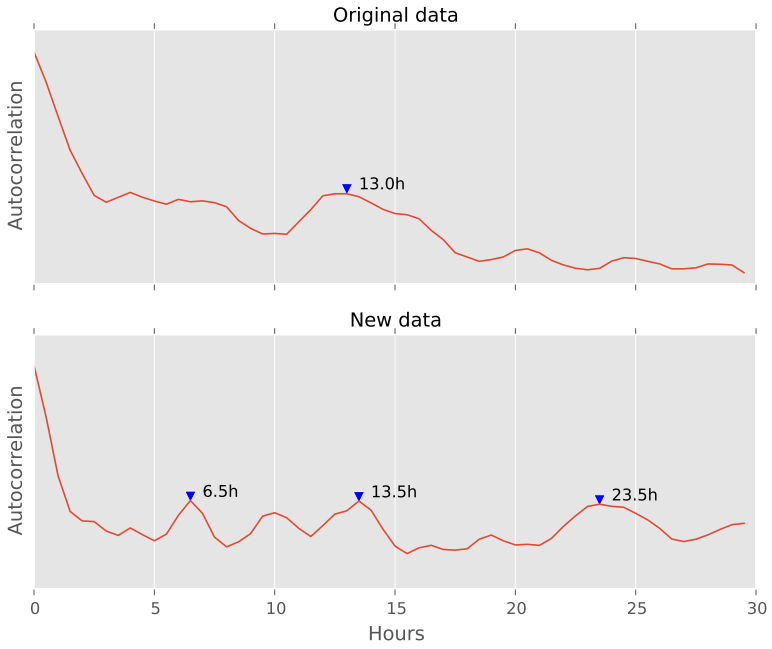


Figure 3.8: Detail of the autocorrelation for video *e*. A peak at ∼12 hours indicates an ultradian periodicity on the original data. Peaks at ∼24 and ∼12 hours on the new data indicate circadian and ultradian periodicities respectively.

3.5 Conclusions

Although according to the original publication the previous manual methodology was enough to detect the circadian rhythms, part of the information about the behavior is lost (e.g., if the lobster moves without crossing the middle of the tank, it is not taken into account). The proposed automatized methodology allows a precision at frame level and quantifies every single movement of the individual. Therefore, the new system showed not only similar results, but periodicities not found before.

Furthermore, the processing time is reduced from hours of manual inspection to minutes of CPU power.

All these benefits were possible using a standard PC and available free open source libraries, therefore a zero-cost solution. The man-hours needed to implement the full processing chain were also positively affected by reusing these public domain implementations of published algorithms, not "reinventing the wheel", as addressed by Spinellis and Szyperski (2004).

The system could grow in complexity, e.g., showing daily heatmaps to monitor where each specimen spends most of the time, which could lead to

		Periodicity		
Tank ID	Method	24 h	18 h	12 h
a	Orig.	✓		
	New	✓		✓
b	Orig.	✓		
	New	✓	✓	
c	Orig.		✓	
	New	✓		
d	Orig.	✓		
	New	✓	✓	
e	Orig.			✓
	New	✓		✓
f	Orig.	✓		✓
	New		✓	✓

Table 3.2: Allocation of the values on table 3.1 to circadian and ultradian periodicities.

conclusions that would help to improve the monitoring methodology itself. This added processing and data visualization could still be performed using only open source solutions, e.g., the Python plotting library matplotlib (Hunter, 2007).

The automation of the monitoring is therefore contributing to be able to spend more time understanding the data instead of collecting the data.

Regarding how to involve citizens on this research, the motion detection algorithm could be complemented with ground truthing performed by volunteers. The results shown in this work have not been manually verified due to the time requirements this process would need —much longer than the original publication, as we are demanding more detail—. This verification could be split between a pool of volunteers, each one visually analyzing a short segment of video according to some guidelines to keep subjectivity to a minimum.

Another sample case of participation would be the manual inspection of undesired effects (e.g., the bubbles of the water circulation system) that are big enough to be misclassified as the subject of interest. This cases are seen by the system as a big change of the individual's position between two frames, as if the lobster had instantaneously "jumped" from one part of the tank to another. This cases could be tagged as suspicious, and then the problematic frames would then be delivered to volunteers (e.g., posted online) that would easily identify the real position of the lobster and manually select its location. In addition to that, if the data were made public alongside the code, users

with software coding knowledge could try to implement new motion detection algorithms and compete between them for the most robust solution.

Chapter 4

In-situ video-image processing using low cost embedded systems and open source tools

This chapter is based on the previous author's work:

Design of a sensor network with adaptive sampling
Conference talk and proceedings in *iEMSs 2008 Conference (The International Environmental Modelling and Software Society Conference)*, Barcelona (Spain), 2008.

4.1 Introduction

Chapter 3 introduced the use and advantages of automated video-analysis on a laboratory environment. The objective of this chapter is to analyze the benefits of using similar techniques on video-footage from field recordings and in-situ processing, and to be a proof of concept of its feasibility.

As ocean observation systems continue to grow in complexity, management of the rapidly increasing volume of data is a recognized concern (Conway, 2006; Gilbert, 1998). The development of ocean observatories tremendously advances the ability to collect and transmit data, producing unprecedented levels of access to data. However, the ability to interpret this avalanche of data, i.e., to extract meaning from artificial fields of data, has expanded much more slowly (Woods et al., 2002). The challenge has become finding what is informative given the scientific interests and needs in a very large field of available data.

Furthermore, it has been shown (Akyildiz et al., 2002) that effective management of sensor resources has become an important issue on sensor networks. Power and data-transmission bandwidth are limited resources that must be used efficiently. An intelligent in-situ data analysis and reactive behavior can

improve the management of network resources, increasing the quality of the collected data and reducing costs (Pons et al., 2008).

Hence, there is a need for a new generation of computational theories and tools to assist humans in extracting useful, interpreted information (knowledge) from the rapidly growing volumes of digital data. One of the problems to address is mapping low-level data (which are typically too voluminous to easily process and understand) into other forms that might be more compact (e.g., classified data, trends), more abstract or more useful (Fayyad et al., 1996).

On this chapter we propose a solution for a concrete case that transforms a continuous 24 h feed of unprocessed video of the marine soil to a segmented video where only relevant activity is shown. Additional precomputed metadata attached to the video stream, automatically generated or with the help of volunteers in the frame of citizen science, would help scientists filtering the segments according to their needs.

4.2 Materials and Methods

4.2.1 Data description

Not having direct access to a video stream directly from submarine equipment, we simulated this stream using stored footage. This footage was recorded by a submarine infrared 3CCD video-camera mounted on the Real-Time Deep-Sea Floor Permanent Observatory (Iwase et al., 2003), at 1100 m depth off Hatsushima Island, in Sagami Bay (central Japan). The period of the recording lasts one week, from 09-04-1999 to 16-04-1999, in time-lapse mode with a frame each 4 s. During this time there was a constant source of illumination of six white-light lamps.

Videos were stored on VHS videotapes and posteriorly digitized (Aguzzi et al., 2009a). The video format we had access was in AVI container, MPEG-2 Video codec and a resolution of 720 × 480 pixels. A sample is shown in figure 4.1.

4.2.2 Processing

The video image processing described on this section is very similar to the processing performed on chapter 3, using C with the open source library OpenCV (Bradski, 2000; Pulli et al., 2012).

In-situ hardware

Instead of performing the processing on a PC, though, it has been fully executed on a Raspberry Pi. The Raspberry Pi is a credit-card sized computer able to run a full-pledged OS like Raspbian (a GNU/Linux distribution based

Figure 4.1: Sample frame from the digitized VHS footage.

on Debian). The model used was the Raspberry Pi 2 Model B, with a price at the moment of the writing of 35USD.

Because of its form-factor, connectivity capabilities and low power consumption, it is gaining popularity as scientific in-situ processing platform. For example:

- As a Wireless Sensor Network (WSN) and SensorWeb node. Vujovic and Maksimovic (2014) consider that the Raspberry Pi brings the advantages of a PC to the domain of sensor networks:

  > "It is the perfect platform for interfacing with a wide variety of external peripherals. It remains an inexpensive computer with its very successfully usage in sensor network domain and diverse range of research applications."

- Neves and Matos (2013) developed an autonomous driving system capable of following another vehicle and moving along paths delimited by colored buoys. A pair of webcams was used and, with an ultrasound sensor, they also implemented a basic frontal obstacle avoidance system.

- As a platform for monitoring water level, velocity and rain intensity (Lanfranchi et al., 2014). A large number of sensors provided spatial patterns and temporal evolution and real-time information for decision-making.

- On field of citizen science, it was used by Wrigley (2014) as a low-cost, mobile device that records atmospheric conditions (temperature,

Figure 4.2: Raspberry Pi with an attached camera module. Image from Adafruit Industries (CC BY-NC-SA 2.0).

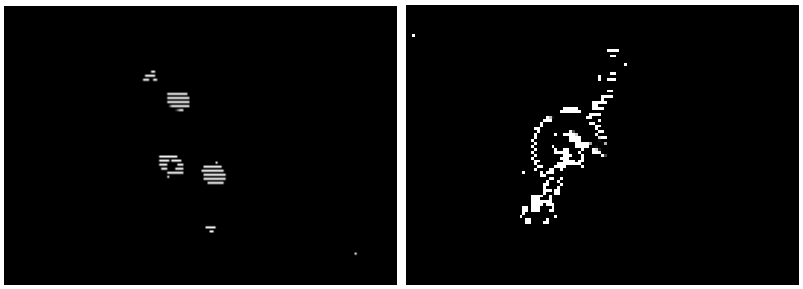
barometric pressure, luminosity, etc.) as well as a means of estimating river flow using the Raspberry Pi camera module (figure 4.2).

- Also regarding citizen science, Gordienko et al. (2015) claim that the Raspberry Pi allows to go from the passive "volunteer computing" to other volunteer actions like "volunteer measurements" under guidance of scientists.

Processing chain

The footage is first subjected to a preprocessing of masking, so that certain parts of the image will not be processed (e.g., the parts with the changing overlay time and date). Next, the image is transformed from the RGB color space to the HSV color space (Smith, 1978). This transformation improves the performance of the later step of background subtraction mentioned below. Empirically, most of the variation in background tends to be along the brightness axis, not the color axis (Bradski and Kaehler, 2008).

Histogram equalization, a method of contrast adjustment using the image's histogram, was studied as a part of the preprocessing. Histogram equalization is an old mathematical technique; its use in image processing is described in various textbooks (Jain, 1989; Russ, 2011). In our case, the equalization modified the values of hue and saturation in a not homogeneously across the



(a) Random objects of non-interest. (b) Contour of detected snail.

Figure 4.3: Zoomed samples of artifacts (horizontal lines) produced during the digitization of the VHS tape, affecting the background subtraction method.

image, impacting the classification algorithm used at the end of the processing chain. Therefore, it was discarded and is not used in this study.

With the aim to identify when fauna appear on the image, the next step is the implementation of a background subtraction method. In this case, instead of using the codebook background subtraction explained on page 14 and used on the previous chapter, the less computer intensive average background method was used (Toyama et al., 1999; Bradski and Kaehler, 2008). This method basically learns the average and standard deviation of each pixel as its model of the background for each channel (hue, saturation and value). In our case, for each frame to process, the model is updated with the previous 100 frames (i.e., a buffer of images is needed for this technique). Then the frame is compared to the model: if the pixel values are out of any of the thresholds of the model, the pixel is considered foreground. Otherwise, it is classified as background.

After binarizing the image using background subtraction, we observed that because of the digitization of the VHS tape, there were artifacts in the form of horizontal lines. Additionally, in many cases we were obtaining open contours instead of filled blobs. These effects can bee seen on figure 4.3.

These artifacts are compensated on the binarized image by applying a filling algorithm proposed by Martin Cirera et al. (2010). A 3×3 pixels window sweeps the image, pixel by pixel. If two pixels on opposite borders are white, and they would form a straight line (horizontal, vertical or diagonal) if the center pixel was also white, then the center pixel is painted white if it wasn't.

Next, we apply a contour finding algorithm (Suzuki et al., 1985), and we keep track only of the objects with an area bigger than a given threshold (defined empirically by trial and error) to discard most of the noise produced by marine snow (Alldredge and Silver, 1988).

For each object, the corresponding HSV region is extracted from the original image. Then, these HSV objects are classified with the following method.

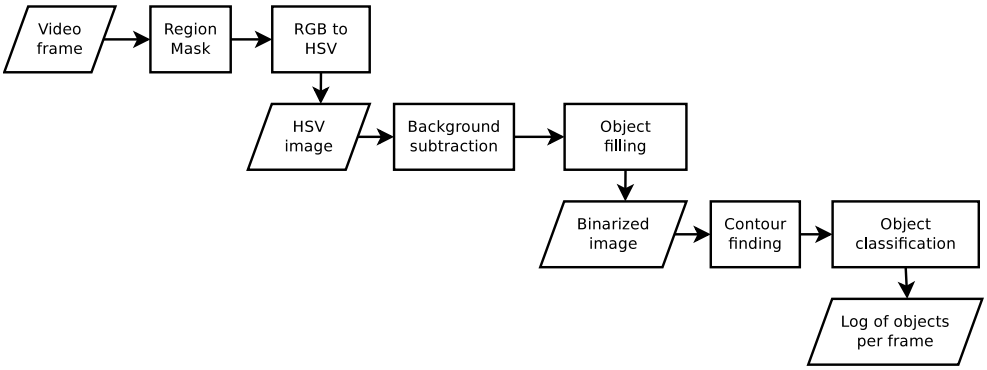


Figure 4.4: Complete video processing chain.

Three different species of interest where manually identified before the processing. Three sample images for each species were stored as reference. During the processing, each detected object is matched against the stored reference images comparing the histograms using the χ^2 distance as measure of similarity (Swain and Ballard, 1991; Schiele and Crowley, 1996).

Given that both histograms of each pair being compared have the same n bins, and representing them as $x = [x_1, ..., x_n]$ and $y = [y_1, ..., y_n]$, the χ^2 distance d is defined as $d(x, y) = \sum((x_i - y_i)^2/(x_i + y_i))/2$.Note that $d(x, y)$ is symmetric with regards to x and y. Before performing this calculation, both histograms need to be normalized, i.e., their entries sum up to one.

The object is labeled as the species with the best match if the similarity measure surpasses a given threshold. If it does no surpass the threshold, it is labeled as unknown species. This information is logged on a text file, associating frame number with the species detected in the frame. The full processing chain is shown in figure 4.4.

4.3 Results

The Rasberry Pi was able to process the video at real-time speed. Figure 4.5 shows the result of applying the motion detection and contour finding algorithms to two sample frames. The three different species included on the classificator can be seen appearing on these two frames. Table 4.1 shows the results of the classification of 15 sample objects, according to the last part of section 4.2.2. Each species has been correctly labeled in this sample.

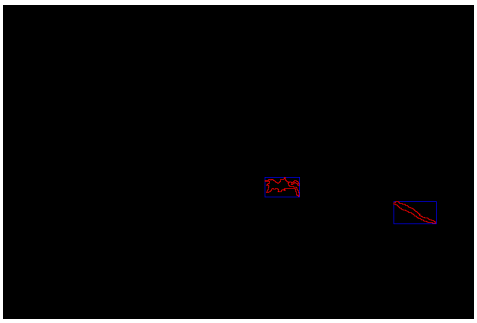
4.4 Discussion

The results on the previous section set the first step on demonstrating the viability of setting up an in-situ video processing system. The Raspberry Pi,

Table 4.1: χ^2 scores from comparing the histogram of 5 sample images of each specie against the reference histograms. The bold value of each row is the best (lower) score.

Detected object		**Reference**								
		Species a			Species b			Species c		
Species	Sample	Image 1	Image 2	Image 3	Image 1	Image 2	Image 3	Image 1	Image 2	Image 3
a	1	0,45805	0,44376	**0,42728**	1,60422	1,52003	1,05533	1,44615	1,02159	0,83716
	2	0,42344	**0,40849**	0,44036	1,53000	1,25094	0,84672	1,35711	0,65622	0,57177
	3	0,17738	**0,16318**	0,23285	1,68422	1,30132	1,30132	1,55978	0,62961	0,31327
	4	**0,11171**	0,21448	0,27987	1,59808	1,20672	0,78390	1,46461	0,48226	0,17464
	5	**0,08229**	0,11881	0,18077	1,61930	1,29578	0,82791	1,47898	0,59152	0,29313
b	1	1,41048	1,45520	1,49658	0,16093	**0,15275**	0,40369	0,20917	0,60121	1,37942
	2	1,76400	1,80443	1,80834	**0,08903**	0,39292	0,58969	0,32553	0,98849	1,82720
	3	1,52205	1,57283	1,57642	**0,11118**	0,34666	0,41285	0,23989	0,78366	1,58932
	4	1,34029	1,39845	1,49569	0,33140	**0,05939**	0,59444	0,56466	0,52244	1,10927
	5	1,25461	1,32527	1,43053	0,34527	**0,17603**	0,44992	0,39022	0,39471	1,05461
c	1	1,22529	1,30097	1,35902	0,37253	0,34858	0,18156	**0,13155**	0,37505	1,14680
	2	0,99953	1,12211	1,20348	0,76218	0,48700	0,25576	0,40571	**0,16716**	0,76267
	3	1,85484	1,87573	1,88761	0,61877	0,86279	0,71357	**0,16580**	1,04117	1,88787
	4	0,90747	1,02269	1,09862	0,85420	0,56482	0,27490	0,45616	**0,14419**	0,66823
	5	0,71288	0,83951	0,94636	0,93847	0,60616	0,32865	0,70044	**0,02990**	0,39443

(a) Sample frame with two species.



(b) Detected contour of the species on subfigure (a).

(c) Sample frame with two species.

(d) Detected contour of the species on subfigure (b).

Figure 4.5: Two sample frames and the contour of the species detected in them with the implemented processing chain.

as seen on section 4.2.2, can be used on underwater deployments. The video file used as input could be replaced by a live feed, as both the software and the hardware allow this possibility.

More complex video processing techniques, like the ones used by Aguzzi et al. (2011), could be ported to the Raspberry Pi hardware, as implementations in C with OpenCV are less computationally demanding than the Matlab equivalent (Matuska et al., 2012).

This study has not addressed the analysis of the log relating detected species with frames, but it is a feature that opens new research lines:

- In the case systems with limited resources regarding data storage or transmission bandwidth, the motion detection or even specific species detection could be used as a decision factor between keeping/transmitting the data or discarding it.

- On a video archiving scenario —independently of if we use the full stream or just discontinuous segments— the log could be transformed to meta-

data to help on the manual inspection of the videos. For example, software would allow to navigate and filter through the video archive according to the parameters chosen by the biologist, such as videos featuring a specific species. An existing system with the same philosophy has been documented by Schlining and Stout (2006).

- The metadata commented on the previous point could be, additionally, a source for generating statistics and characterizing fauna behavior, similar to the processing from chapter 3.

4.5 Conclusions

This chapter enunciated in-situ processing as one of the solutions to the increasing oceanographic data acquisition capabilities. The analysis and filtering of data on the same point where data is collected allow and optimization of resources both technical and human.

The technical feasibility of this philosophy has been demonstrated with one of the more complex sources of data: a video stream. Existing open source libraries and off-the-shelf low-cost hardware are already available to build such systems. This study implemented a simulation of a submarine node successfully analyzing and categorizing video in real-time on embedded hardware.

The future possibilities that this example of in-situ processing opens have been discussed (section 4.4). As stated in previous chapters, the field of oceanographic sciences can greatly benefit of adopting the technology already in use in other areas. With this technology, data can be turned into information —and therefore into knowledge— easier and with less resources.

With the mentorship of scientists, citizens living close to shallow water masses could build and deploy their own amateur observational underwater nodes with a limited investment. Sharing the captured preprocessed footage with other users and classifying it interactively as commented on section 4.4 would be an unprecedented source of data for characterizing the ecosystem that would complement the current scientific and systematic approach.

Chapter 5

Biological time series forecasting based on open source tools

This chapter is based on the previous author's work:

Early-Warning System Based on Decision Trees for Blooms of Harmful Algae
Research article submitted to *PLOS ONE*, October 2015.

5.1 Introduction

The objective of this chapter is to introduce open source machine learning software as a method to improve biological observations. In this case, using data from an existing monitoring field campaign.

Aquatic ecosystems are characterized by an extraordinary mix of human activities, e.g.: tourism, fishing and aquaculture. Algae are a critical part of aquatic ecosystems and, like land plants, capture the sun's energy and support the food web that includes fishes and shellfish. Harmful algal blooms (HABs) are related to negative impacts to other organisms via production of toxins, mechanical damage or oxygen depletion. HABs are often associated with large-scale marine mortality events and with various types of shellfish poisonings (Backer et al., 2003). Some algae produce powerful toxins which cause illness or death in humans and other organisms. Other algae are not toxic, but affect whole ecosystems by forming large mats that can have an adverse effect on corals, seagrasses, and other organisms living on the sea-bottom.

The assessment of impacts of HABs on whole ecosystems over long periods of time involves the complex processing of large databases, covering sometimes wide geographical areas. To secure the funding needed for data collection in

those cases it is important to look at new cost-effective ways of obtaining and processing environmental monitored data.

Existing bloom forecast systems use as inputs a combination of in-situ hydrographic, water quality, meteorological, biomass and/or taxonomic data. These approaches rely therefore on the deployment of multiple sensing devices or systematic field campaigns. A less costly approach, but restricted to water masses with an extension of the order of kilometers because of its low spatial resolution, is the use of remote sensing data.

A detailed listing of input descriptors used by existing literature can be seen on table 5.1. Most forecasting systems use remote-sensing data (Allen et al., 2008; Stumpf et al., 2009) or a large amount of field measurements (Sivapragasam et al., 2010; Lee et al., 2005), in many cases to feed a specific hydrodynamic model of the environment under study (Zhang et al., 2013; Wong et al., 2009), usually requiring time-consuming processes or costly equipment. In contrast, the system proposed below only requires the input of chlorophyll a and concentration of algae data, and it is not adjusted to a specific environment and its hydrodynamic characteristics. Exporting the proposed methodology to other scenarios would not imply the deployment of complex equipment.

In this chapter we propose a forecast system based on decision trees implemented with open source software. This system uses as input a reduced number of microscopy measurements (one measurement per harmful species). These measurements are already being performed as part of a periodic environment monitoring; no extra infrastructure has been deployed or human resources have been needed for the acquisition of data.

5.2 Materials and Methods

This work explores patterns of harmful microalgae variability in the *Alfacs* bay (Spain), specifically, temporal patterns in series of harmful phytoplankton species abundances sampled from a fixed station, using automatic-classification techniques. Decision trees are used to infer the probability of having a toxic HAB on the following one, two or three weeks, respectively.

5.2.1 Data description

The data used in this work were collected through the monitoring program on water quality of shellfish growing areas in Catalonia conducted by the Institute of Agrifood Research and Technology in 1998–2003 and 2006–2010. Water samples were taken approximately every week, and from these samples microalgae species were quantified. Seven dinoflagellate species and one diatom genus (*Pseudo-nitzschia*) were selected for this study, based on their association with shellfish closures in Catalonia. Hydrographic descriptors (temperature, salinity and dissolved oxygen concentration) were also measured

	Input descriptors	Reference												
		a	b	c	d	e	f	g	h	i	j	k	l	m
Taxonomic based	Algal concentration of single species							✓						✓
	Algal concentration of harmful species	✓										✓		
	Algal concentration of non harmful species											✓		
	Algal grow rate				✓									
Water quality	Nutrients (phosphorus, nitrate)		✓	✓	✓				✓	✓	✓	✓		✓
	Water quality (including Chl *a*)		✓	✓	✓	✓			✓	✓	✓	✓		✓
	Biomass concentration (phytoplankton, zooplankton)			✓					✓	✓			✓	
Others	Water flow, turbulence/stratification			✓	✓							✓		
	Meteorology			✓	✓	✓		✓			✓	✓		✓
	Remote sensing			✓			✓	✓					✓	

Table 5.1: Comparison with other studies with respect to input descriptors: a, this study; b, Recknagel et al. (1997); c, Zhang et al. (2013); d, Wong et al. (2009); e, Muttil and Lee (2005); f, Allen et al. (2008); g, Stumpf et al. (2009); h, Whigham and Recknagel (2001); i, Wei et al. (2001); j, Sivapragasam et al. (2010); k, Lee et al. (2005); l, Anderson et al. (2011); m, Chen and Mynett (2004).

using a portable multi-parametric probe. The complete set of descriptors (or attributes) is therefore as follows:

- Temperature in °C (Temp)
- Dissolved Oxygen in % (O_2)
- Salinity (S)
- Chlorophyll *a* in μg/l (Chl *a*)
- Concentration (cells/L) of algae:
  - *Alexandrium catenella*
  - *Alexandrium minutum*
  - *Dinophysis caudata*

- *Phalacroma rotundatum*
- *Dinophysis sacculus*
- *Protoceratium reticulatum*
- *Protocentrum lima*
- *Pseudo-nitzschia spp.*

The original sampling process was not equally distributed in time. On some weeks the sampling was not performed; on some weeks it was performed more than once. We preprocessed the time series by filling the missing samples using linear interpolation and calculating the mean value on the weeks with multiple samples.

The results of a first exploration phase indicated that the classification is mainly driven by taxon information. Therefore, in a second phase of experimentation, parameters regarding meteorology and hydrography have been discarded in order to make the predictor more efficient, requiring less inputs.

5.2.2 Processing

Data-driven modeling is a fast developing field in hydroinformatics (Solomatine et al., 2008) consisting on finding connections between the system state variables (input, internal, and output variables) without explicit knowledge of the physical behavior of the system. Once the model is trained, it can be tested using an independent data set to determine how well it can be extrapolated to unseen data.

Past comparisons (Recknagel, 1997) between the predictive potential of deductive and inductive learning of phytoplankton models regarding their forecasting of harmful algal blooms showed that the application of deductive models still seem to be restricted by a lack of knowledge, while ad-hoc inductive models prove to be more predictive. Previous studies have demonstrated that predictive models based on microbial and ecological processes in freshwater bodies are useful for developing management responses aimed at reducing the negative consequences of algal blooms to the community (Whigham and Recknagel, 2001; Wilson and Recknagel, 2001; Maier and Dandy, 2000; Wei et al., 2001; Muttil and Chau, 2006). In a case similar to the one discussed in this study, with a time series comprised of biomass of ten dominating algae species and environmental driving variables over twelve years, artificial neural networks were used to predict the succession, timing and magnitudes of algal species, indicating that neural networks can fit the complexity and non-linearity of ecological phenomena apparently to a high degree (Recknagel et al., 1997).

Our early-warning system for HABs forecasting uses observations of phytoplankton species abundances. For each species or genus of harmful phytoplankton, a discrete predictor with three levels of concentration was defined:

Species or genus	Abbreviation	Concentration (cells/L)
Alexandrium catenella	acatenella	1,000
Alexandrium minutum	amin	1,000
Dinophysis caudata	dcaud	500
Phalacroma rotundatum	drotund	500
Dinophysis sacculus	dsacculus	500
Pseudo-nitzschia spp.	psnit	200,000

Table 5.2: Concentration values set as thresholds for defining HABs. Two species were not used to define HABs, but only to predict their occurrence.

- A *threshold level* based on the risk of toxin accumulation in shellfish (Andersen, 1996; Anderson et al., 2001). The threshold values depend on local legislation; the ones used for this study are established for shellfish growing areas in Catalonia (Spain) and are shown in table 5.2. The threshold level ranges from "threshold value − 10%" to "threshold value + 10%". For example, for *Alexandrium catenella*, it would be [900–1100] cells/L.

- A *low level*, which ranges between 0 and the threshold level. For example, for *Alexandrium catenella*, it would be [0–900] cells/L.

- A *high level*, which includes values greater than the threshold level. For example, for *Alexandrium catenella*, values greater than 1100 cells/L.

To improve prediction, two additional predictors, derived from the concentration time series, were defined and used: the trend and the accumulated value. The trend of a parameter is defined as the sign of the derivative of a descriptor over the last two weeks, being positive if the concentration increases and negative if it decreases. The accumulated value is quantified as the integral of a descriptor over the last four weeks, using the composite trapezoidal rule.

The trapezoidal rule is a technique for approximating the definite integral $\int_a^b f(x)dx$. The trapezoidal rule works by approximating the region under the graph of the function $f(x)$ as a trapezoid and calculating its area, following that $\int_a^b f(x)dx \approx (b-a)[(f(a)+f(b))/2]$. Figure 5.1 illustrates this concept. The composite is its repeated application along different segments of $f(x)$, i.e., the concentration time series in our case.

The complete set of descriptors used in this study are listed in section 5.2.2.

With this information, an inductive *learning element* (decision trees) has been used to forecast HABs, defined as surpassing one of the thresholds of table 5.2, one, two and three weeks in advance. The design of the learning element takes into account three major issues:

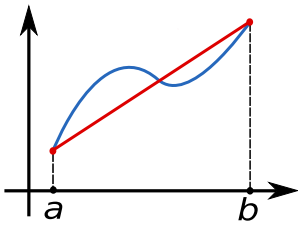


Figure 5.1: Example of application of the trapezoidal rule. The function $f(x)$ (in blue) is approximated by a linear function (in red) between the points a, b .

1. which *descriptors* are to be learned;
2. what *feedback* is available to learn these descriptors;
3. what *representation* is used for the descriptors.

The *descriptors* to be learned are the concentration levels of algae and recent trend and integral of these concentrations. The *type of feedback* available for learning determined the nature of the learning problem that the system faces: *supervised* learning, which involves learning a *function* from examples of inputs and outputs. The system learns a function from observations of phytoplankton species abundances to a boolean output (whether there is a HAB). Finally, the *representation of the learned information*, propositional logic, plays a very important role in determining how the learning algorithms work.

The last major factor in the design of the learning system was the *availability of prior knowledge*. The system begins with no knowledge about what it is trying to learn. It has access only to the examples in the data series.

Learning decision trees

In this study, an algorithm for deterministic supervised learning is given as input the correct value of the unknown function for particular inputs and tries to recover the unknown function or something close to it. More formally, we say that an *example* is a pair *(x, f(x))*, where x is the input and *f(x)* is the output of the function applied to x. The task of *pure inductive inference* (or *induction*) is this: given a collection of examples of f, return a function h that approximates f.

Decision tree induction is used in this study, being one of the most successful forms of learning algorithm and being the decision tree representation very natural for humans. The *decision tree* takes as input a situation described by a set of *descriptors* (a vector of attribute values) and returns a *decision*: a single, predicted output value for the input. The input descriptors have discrete (concentration level: threshold level, low level, high level; trend:

positive, negative) and continuous (accumulated concentration) values. The output is discrete and has exactly two possible values; therefore this a case of *classification* learning (the system is learning a discrete-valued function), and, specifically, of *Boolean* classification, wherein each example input is classified as true (a *positive* example) or false (a *negative* example).

The decision tree reaches its decision by performing a sequence of tests. Each internal node in the tree corresponds to a test of the value of one of the input descriptors, and the branches from the node are labeled with the possible values of the descriptor. Each leaf node in the tree specifies the value to be returned by the function if that leaf is reached.

The aim here is to learn a definition for the **goal predicate** *Bloom*. We set this up as a learning problem and state what descriptors are available to describe examples in the domain as part of the input, which are the ones on the following list:

1. *Alexandrium catenella* concentration (acatenella)
2. *Alexandrium minutum* concentration (amin)
3. *Dinophysis caudata* concentration (dcaud)
4. *Phalacroma rotundatum* concentration (drotund)
5. *Dinophysis sacculus* concentration (dsacculus)
6. *Protoceratium reticulatum* concentration (preticul)
7. *Protocentrum lima* concentration (prlim)
8. *Pseudo-nitzschia spp.* concentration (psnit)
9. *Alexandrium catenella* concentration derived over two weeks (acatenella_s)
10. *Alexandrium minutum* concentration derived over two weeks (amin_s)
11. *Dinophysis caudata* concentration derived over two weeks (dcaud_s)
12. *Phalacroma rotundatum* concentration derived over two weeks (drotund_s)
13. *Dinophysis sacculus* concentration derived over two weeks (dsacculus_s)
14. *Protoceratium reticulatum* concentration derived over two weeks (preticul_s)
15. *Protocentrum lima* concentration derived over two weeks (prlim_s)
16. *Pseudo-nitzschia spp.* concentration derived over two weeks (psnit_s)
17. *Alexandrium catenella* concentration integrated over four weeks (acatenella_i)
18. *Alexandrium minutum* concentration integrated over four weeks (amin_i)

19. *Dinophysis caudata* concentration integrated over four weeks (dcaud_i)
20. *Phalacroma rotundatum* concentration integrated over four weeks (drotund_i)
21. *Dinophysis sacculus* concentration integrated over four weeks (dsacculus_i)
22. *Protoceratium reticulatum* concentration integrated over four weeks (preticul_i)
23. *Protocentrum lima* concentration integrated over four weeks (prlim_i)
24. *Pseudo-nitzschia spp.* concentration integrated over four weeks (psnit_i)

A decision tree for this domain is shown in figure 5.2.

In the task of finding the optimal decision tree, heuristics are used to find solutions in an acceptable time, but they are not guaranteed to be optimal. There are many such heuristics for deciding the sequence of tests and the specification of each test: C4.5 (Quinlan, 1993), C5.0 (Quinlan, 2004), the Gini index (Gini, 1912), and others (Mitchell, 1997).

For being the one that produced the best classification results in preliminary tests, the Gini index is used in this study. In short, the Gini index is a measure of the inequality of a distribution, i.e., it is a measure of statistical dispersion. The Gini index can be computed by summing the probability of each item being chosen times the probability of a mistake in categorizing that item. It reaches its minimum (zero) when all cases in the node fall into a single target category.

The forecasting system is composed of six different decision trees (implemented in Python using the Orange data-mining toolbox (Demšar et al., 2013)), which predict bloom events one, two and three weeks in advance, respectively. Three cases use the full set of descriptors, and the other three a subset of only the concentration levels (descriptors 1–8 from the list above). This subset of descriptors are used to show the effect of the additional derived descriptors on the system performance. Furthermore, the performance of these decision trees is also compared to classifiers using the *K nearest neighbors* (KNN) technique with the same data input, totaling twelve cases summarized on table 5.3.

The decision trees learning configuration is:

- Data subsets with less than 10 instances are not split any further.
- Induction stops when the proportion of majority class in a node exceeds 85%.
- The is a bottom-up post-pruning by removing the subtrees of which all leaves classify to the same class.

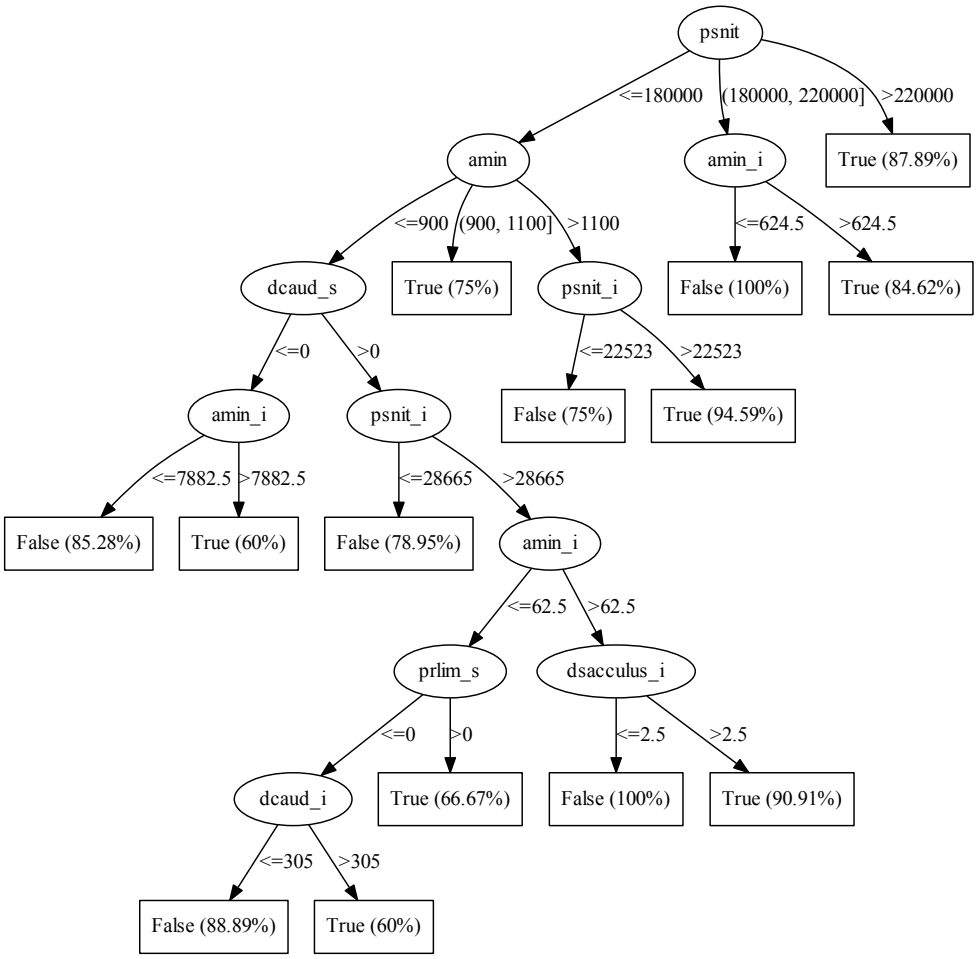


Figure 5.2: Decision tree for case *b*. The percentage associated to the prediction of each leave corresponds to the confidence level. Notice that this particular tree only uses the *amin*, *psnit*, *dcaud_s*, *prlim_s*, *amin_i*, *dcaud_i*, *dsacculus_i* and *psnit_i* descriptors, in effect considering the rest to be irrelevant.

Case	Learning system		Prediction			Input		
	Decision tree	KNN	One week	Two weeks	Three weeks	Single data point	Integral	Derivative
a	✓		✓			✓		
b	✓		✓			✓	✓	✓
c		✓	✓			✓		
d		✓	✓			✓	✓	✓
e	✓			✓		✓		
f	✓			✓		✓	✓	✓
g		✓		✓		✓		
h		✓		✓		✓	✓	✓
i	✓				✓	✓		
j	✓				✓	✓	✓	✓
k		✓			✓	✓		
l		✓			✓	✓	✓	✓

Table 5.3: Characteristics of the cases studied. The *Prediction* column indicates the advance in forecast time. The input to each system can be: the concentration level of the current week (single data point), the integral of the concentrations of the last four weeks, and the derivative of the concentration of the last two weeks.

The proportion between classes on the original dataset was of 28.7% of positive samples and 71.3% of negative samples. Therefore, it is a case of class imbalance in binary classification (Japkowicz, 2000). Imbalance has a serious impact on the performance of classifiers. Learning algorithms that do not consider class imbalance tend to be overwhelmed by the majority class and ignore the minority class Chawla et al. (2004). Two typical approaches followed to address the class imbalance problem are:

Over-sampling: Increasing the number of minority instances by replicating them. This results in a larger dataset which retains all the original data but may introduce bias. As the size increases, however, it can impact computational performance as well (Ling and Li, 1998).

Under-sampling: Extracting a smaller set of the majority instances and keeping the minority. This results in a smaller dataset where the distribution between classes is closer. However, data has been discarded that could have been valuable (Kubat and Matwin, 1997).

We followed the under-sampling approach, by taking out a 50% of the negative samples for the training and testing of the systems. The proportion on the new dataset was of 44.6% of positive samples and 55.4% of negative samples.

5.3 Results

Twelve different cases of the predictor have been studied (table 5.3 shows their characteristics) and the following indicators have been calculated:

- Accuracy: $(\text{TPs} + \text{TNs})/(\text{TPs} + \text{TNs} + \text{FPs} + \text{FNs})$
- Precision: $\text{TPs}/(\text{TPs} + \text{FPs})$
- Sensitivity: $\text{TPs}/(\text{TPs} + \text{FNs})$
- Specificity: $\text{TNs}/(\text{TNs} + \text{FPs})$

where TPs are *true positives* (actual bloom cases that have been predicted successfully); TNs are *true negatives* (actual non-bloom cases that have been predicted successfully); FPs are *false positives* (actual non-bloom cases that have been predicted as bloom cases); and FNs are *false negatives* (actual bloom cases that were not predicted).

The predictors have been tested with a leave-one-out cross-validation (Stone, 1974). Cross-validation is a model validation technique for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction and one wants to estimate how accurately a predictive model will perform in practice, as is our case. In a prediction problem, a model is usually given a dataset of known data on which training is run (training dataset), and a dataset of unknown data against which the model is tested (testing dataset). Leave-one-out cross-validation is a cross-validation where given a dataset of N points, the predictor is trained on all the data except for one point and a prediction is made for that point, repeated N separate times. The average error is computed and used to evaluate the predictor. It is known (Vapnik and Chapelle, 1999; Chapelle et al., 2002) that the leave-one-out procedure gives an almost unbiased estimate of the expected generalization error. The results are shown in table 5.4.

The decision tree (*a, b, e, f, i, j*) performs better than the equivalent KNN (*c, d, g, h, k, l*) with respect to accuracy and also, in the majority of the cases, with respect to precision, sensitivity and specificity.

In the cases of one-week forecasting, the systems using only the concentration value as input (*a, c*) performed better than the systems using all the descriptors (*b, d*). This can be generalized to the majority of the cases also for two-week and three-week forecasting.

5.4 Discussion

The objective of the forecasting efforts of this study is to be able to predict the maximum number of bloom cases, minimizing at the same time the number of false negatives, so that the system can be trusted. From a bay management

Case	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)
a	**83**	**85**	**76**	**89**
b	**82**	**83**	**75**	87
c	**82**	**83**	**74**	88
d	76	73	73	78
e	78	80	67	87
f	79	81	68	87
g	77	82	63	**89**
h	74	71	71	77
i	78	81	65	88
j	78	80	67	86
k	53	35	06	**90**
l	69	66	65	73

Table 5.4: Performance of the system. Average data are shown. Performance is calculated using leave-one-out cross-validation. Bold text highlights best results for each column.

perspective, the priority is not to miss blooms, even at the cost of increasing false negatives. Therefore, the most relevant indicator, to evaluate the performance of the system, is the sensitivity, as it is (inversely) related to how many true blooms the system is missing. The second most relevant indicator is the precision, as it represents the proportion between true positives and false positives.

Overall, in one-week forecasting, the best system is the one using decision trees and the concentration level of the current week (case a), with an accuracy of 83%, precision of 85%, sensitivity of 76% and specificity of 89%. In two-week forecasting, the best system is the one using KNN and the concentration level of the current week (case g), with an accuracy of 77%, precision of 82%, sensitivity of 63% and specificity of 89%. In three-week forecasting, the best system is the one using decision trees and the the full set of descriptors (case j), with an accuracy of 78%, precision of 80%, sensitivity of 67% and specificity of 90%.

As expected, the quality of the prediction degrades in time (see figure 5.3), therefore a trade-off exists between the quality of the prediction and the time at disposal to take decisions.

A fact to highlight is that the majority of cases do not benefit from using extra descriptors (the derivative and the integral of the concentration). This indicates that a better knowledge-representation is needed and, if found, it will improve the results.

Regarding the most relevant performance metric, sensitivity, the proposed systems achieve rates of 65–76%, higher than the success rate of 62% at-

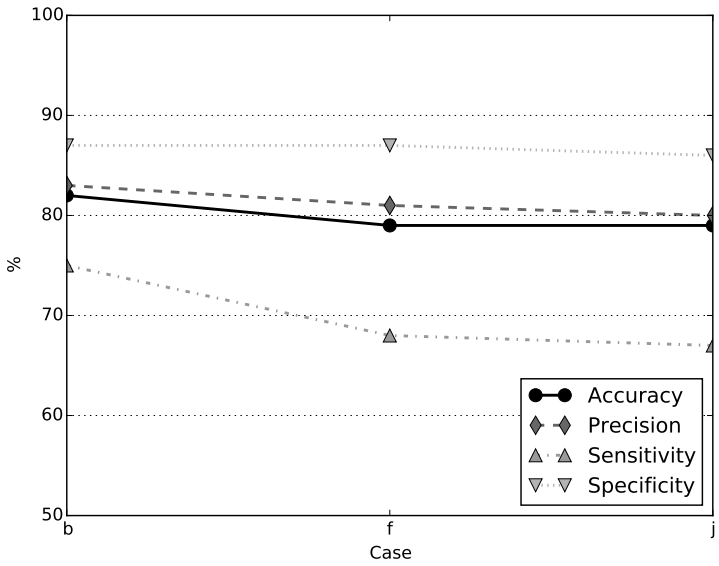


Figure 5.3: Performance comparison of the predictor with data from one week before (case b), two weeks before (case f) and three weeks before (case j) using the same input descriptors.

tributed to regional models (Wong et al., 2009).

Precision, the second most relevant indicator, achieves rates of 80%–85%, comparable to the 85% rate of the study cited above these lines.

5.5 Conclusions

The forecast of algal-bloom occurrence using machine-learning techniques, implemented exclusively with open source software libraries, has been investigated for a bay in the North Western Mediterranean. The K nearest neighbors method (KNN), decision tree algorithms and different knowledge representations have been applied to a database of cases spanning a decade. Blooms are predicted one, two and three weeks in advance. It has been found that the best results in predicting algal-bloom occurrence are obtained: in one-week forecasting, using decision trees and phytoplankton cell abundances of the current week; in two-week forecasting, using KNN and phytoplankton cell abundances of the current week; in three-week forecasting, using decision trees and phytoplankton cell abundances, their trend and their accumulated value. Decision trees give better results when compared to KNN in the majority of the cases.

Data used in this study refer to a weekly sampling-frequency and were

sufficient to achieve a 65–76% of success in precision, which quantifies how many bloom events the system is missing. Precision achieve rates of 80%–85%, which quantifies how many of the detected blooms are not false alarms.

The computational data-processing time for the forecast is of the order of minutes on a common desktop computer (e.g., Intel Core 2 Duo 1.6 GHz processor with 2 GB of RAM), but the previous water sampling and analysis to determine the phytoplankton abundances take two to three days, leaving only a few days to organize a preventive action in the case of a forecast one week in advance.

Current monitoring programs rely on manual optical microscopy for phytoplankton identification, but new imaging systems are being developed (Campbell et al., 2010; Sosik and Olson, 2007; Dubelaar and Gerritzen, 2000), which are capable of unattended, long-duration deployment and produce high-quality images that allow many phytoplankton cells to be identified with respect to genus or even species, with automated image-classification's accuracy comparable to that of human experts. Replacing the manual sampling methodology with a faster, automated sampling system would provide a longer reaction window. This automated-sampling technology is foreseen to become cheaper, more manageable, and easier to operate (Islam et al., 2011), and therefore suitable to be deployed at a large scale.

Future work includes feeding the system with data coming from new, automated flow cytometers (laser-based, biophysical instruments employed in cell counting and cell sorting, which work by suspending cells in a stream of fluid and passing them by an electronic detection apparatus). These cytometers are capable of rapid, unattended analysis of individual plankton cells for long periods of time, allowing to set-up automatic monitoring sensor-networks.

Future work also includes the study of how much the results would improve with a higher sampling-frequency.

Regarding how to involve citizens on this research, if the data were made public alongside the code, as in the project *Citclops Data Explorer* (EyeOnWater, 2015), users with software coding knowledge could try to implement new forecasting algorithms and compete between them for the most robust solution.

Additionally, this project could be coupled with other initiatives focused on phytoplankton monitoring. Zooniverse's Plankton Portal (Borne and Team, 2011), a project where volunteers help marking images of plankton taken by underwater imaging systems, could make use of the bloom forecasting to know when "more interesting" data (i.e., with a higher phytoplankton concentration) is going to happen.

Chapter 6

Bio-optical time series forecasting based on DIY technologies: a modeling approach

This chapter is based on the previous author's work:

Monolithic spectrometer for environmental monitoring applications
3rd prize in student poster competition in *IEEE/OEE Oceans Conference and Exhibition, Oceans'07*, Aberdeen Scotland, 2007.

Low cost hyperspectral device suitable for monitoring sensor networks
Conference talk and proceedings in *1st MARTECH International Workshop On Marine Technology*, Vilanova i la Geltrú (Spain), 2007.

Low cost hyperspectral sensors: potential applications for characterization of multi-scale ocean processes
Conference talk and proceedings in *1st EOS Topical Meeting on Blue Photonics® - Optics in the Sea*, Aberdeen (Scotland), 2009.

6.1 Introduction

On chapter 3 we showed how open source software was used to tackle the automatic analysis of video images. Chapter 4 expanded this concept by adding the use of low-cost hardware for in-situ data processing. Chapter 5 addressed the forecasting of harmful algal blooms applying also open source software. The current chapter, following the same structure, introduces again the use of low-cost hardware for an issue similar to the previous chapter.

Color is probably the most informative cue for object recognition and classification in natural scenes (Rzhanov et al., 2015). Optical sensors have proven useful for pollution detection and biological studies, in addition to prediction of underwater visibility and water depth for navigation purposes and naval operations (Dickey, 2004). They have significant advantages compared to conventional sensor types, in terms of their properties (Forrest et al., 1996). Some of the advantages of optical over non-optical sensors are:

- Greater sensitivity
- Electrical passiveness
- Ease of miniaturization
- Light weight
- Freedom from electromagnetic and other common interferences
- Reduced maintenance
- Ruggedness
- Feasibility of distributed sensing

Hyperspectral sensing is the analysis of signals using a large number of optical channels —corresponding to spectrum intervals—. It is distinguished from multispectral sensing by employing significantly more than the typical 3–8 channels. Capturing the same object on many bands of the spectrum to generate a data cube can reveal objects and information that more limited scanners cannot pick up. Hyperspectral data analysis has shown to be a reliable technique to identify several water components (Louchard et al., 2002; Lee and Carder, 2002).

In the past, hyperspectral sensors were mainly used in laboratories or on expensive mobile platforms due to their big size and high cost, which reduced their applicability for in-situ measurements. Miniaturization has increased the portability of this type of sensors and has provided additional advantages: price and power requirements have also been reduced considerably.

In this chapter we present some preliminar results, as proof of concept, of how the manual water sampling and algal concentration measurements from chapter 5 could be replaced by low-cost do-it-yourself hardware with a miniaturized hyperspectral sensor. First we simulate the optical data that would be captured by a system of these characteristics. Next, this data is processed using a decision tree similarly to section 5.2.2 to estimate if the concentration from a harmful algal species exceeds an alarm threshold.

6.2 Materials and Methods

6.2.1 Data description

The logistics of deploying and maintaining an underwater hyperspectral sensor for a duration similar to the time series from chapter 5 were out of the scope of this thesis. Instead, the generation of data relied on the use of the radiative transfer numerical model Hydrolight/Ecolight 5.0 (Mobley and Sundman, 2008). A radiative transfer model is a model implementing the Radiative Transfer Equation (RTE), which in our case describes the change of the light field under water due to absorption, emission, and scattering processes (Mobley, 1994).

Using this model and the algal concentrations from chapter 5, we simulated the measurements of an hyperspectral sensor. The mesurand chosen was the vertical diffuse attenuation coefficient for downwelling irradiance (K_d). K_d is an important optical property often used in determination of photosynthetic and biological processes in the water column (Marra et al., 1995; McClain et al., 1996). It is an apparent optical property (AOP) that depends on both the medium and geometric structure of the ambient light field. However, as shown in many observations, it is often insensitive to environmental effects except for extreme conditions (Simon and Shanmugam, 2013). The sampling point was simulated for a depth of 0.5 m and the wavelengths of 320, 360, 400, 440, 480, 530, 615, 680, 715, 750, 780, 827.5 and 865 nm. Sample simulated K_d spectra are shown in figure 6.6.

The radiative transfer model had a built-in library of optical properties for different algal groups. We aggregated the concentration levels of each algal species from the monitoring progam mentioned on section 5.2.1 based on its algal group. Hence, we obtained a dataset of 4 variables (the concentration levels of 4 algals groups) from a dataset of 73 variables (the concentration levels of 73 algal groups):

Cryptophyceae: *Cryptomonads.*

Diatoms: *Cylindrotheca closterium, Asterionellopsis glacialis, Asterionella formosa, Thalassionema nitzschioides, Thalassionema frauenfeldii, Lioloma pacificum, Cerataulina pelagica, Chaetoceros curvisetus, Chaetoceros didymus, Chaetoceros lorenzianus, Chaetoceros peruvianus, Chaetoceros pseudocurvisetus, Chaetoceros socialis, Leptocylindrus mediterraneus, Detonula pumila, Ditylum brightwellii, Guinardia flaccida, Hemiaulus sinensis, Hemidiscus cuneiformis, Planktoniella sol, Rhizosolenia cf. imbricata, Guinardia delicatula, Dactyliosolen fragilissimus, Rhizosolenia robusta, Rhizosolenia setigera, Guinardia striata, Skeletonema costatum, Stephanopyxis turris, Thalassiosira eccentrica, Thalassiosira rotula.*

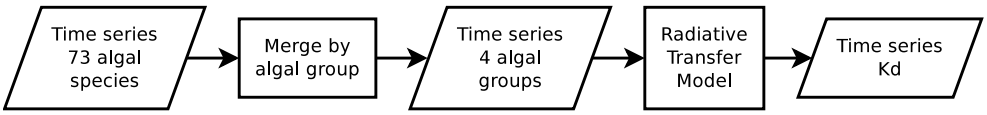


Figure 6.1: Processing chain to generate the time series of K_d.

Dinophyceae: *Alexandrium minutum, Alexandrium tamarense, Ceratium furca, Ceratium fusus, Protoceratium reticulatum, Ceratium pentagonum, Ceratocorys horrida, Dinophysis ovum, Dinophysis acuta, Dinophysis caudata, Dinophysis rotundata, Dinophysis sacculus, Dinophysis tripos, Gonyaulax polyedra, Gonyaulax polygramma, Gymnodinium sanguineum, Gyrodinium spirale, Leptodiscus medusoides, Mesoporos perforatus, Noctiluca scintillans, Oxyphysis oxytoxoides, Oxytoxum longiceps, Podolampas spinifer, Podolampas bipes, Podolampas palmipes, Prorocentrum lima, Prorocentrum mexicanum, Prorocentrum micans, Prorocentrum minimum, Prorocentrum triestinum, Protoperidinium divergens, Prorocentrum rostratum, Glenodinium foliaceum, Gymnodinium pulchellum, Alexandrium catenella.*

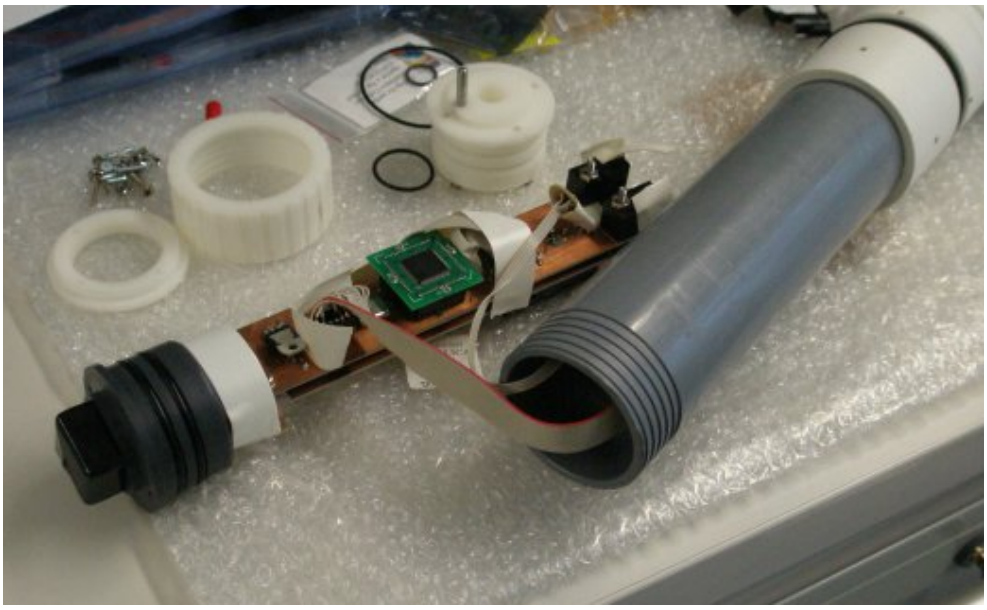
Prymnesiophyceae: *Acanthoica quattrospina, Coronosphaera mediterranea, Emiliania huxleyi, Helicosphaera carteri, Rhabdosphaera clavigera, Syracosphaera pulchra, Umbellosphaera tenuis.*

We fed the radiative transfer model with this new time series. The data corresponding to this complete list of species was available for the period 1998–2003. The bay was characterized as not having any other active optic components (e.g., sediments or organic matter), with an homogeneous water column structure and infinite depth. The objective of this exercise is not to obtain an accurate simulation of the bay, but a first exploration of the feasibility of detecting the presence of harmful algal blooms using low-cost in-situ optical sensors.

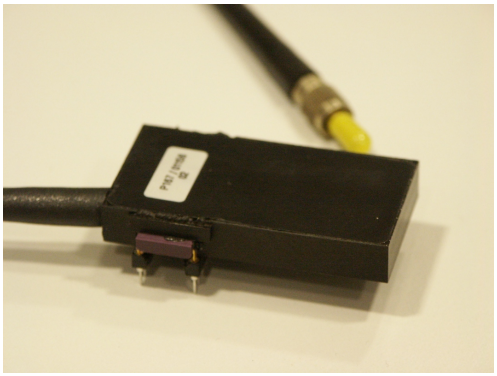
The full processing chain to simulate K_d measurements from concentrations of algal species is shown in figure 6.1.

In-situ hardware

The simulated measurements could be performed with a system similar to the one described by Pons et al. (2007) found in appendix B. This study showed a low-cost custom-made hyperspectral sensing system. The sensor was a CMOS monolithic microspectrometer module of small size ($54 \times 32 \times 9.5$ mm) from *Boehringer Ingelheim microParts GmbH*, governed by a *Microchip* dsPIC 16 bit microcontroller (figure 6.2). The sensor allowed to cover a spectrum range from 200 to 1000 nm with 256 bands, each band converted to a 16 bit value. The sensitivity of the device depended on the integration time (the



(a) The complete system next to a watertight PVC container, to be used as a profiler on a water column.



(b) Detail of the monolithic microspectrometer module.

Figure 6.2: Custom made low-cost hyperspectral system designed and developed by Pons et al. (2007).

longer the time, the more photons collected). The reduced size and low consumption made it suitable for isolated sensing with a high operation autonomy. Furthermore, its low cost allowed the deployment of several devices, enabling a potential sensor network for wide area monitoring.

This system has been compared to commercial alternatives of higher cost designed to be used on a laboratory environment (Torrecilla et al., 2007, 2009). These studies confirmed that this type of microspectrometers are a

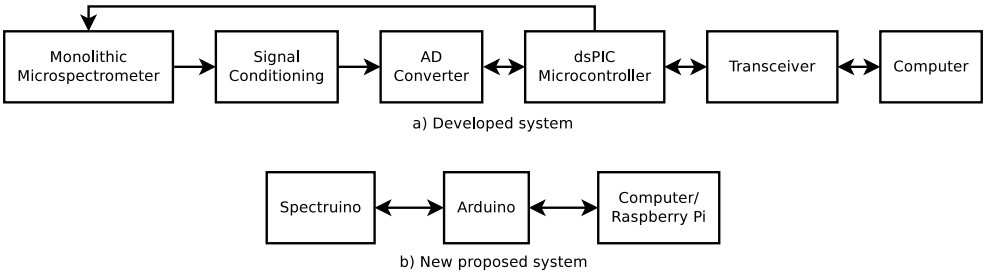


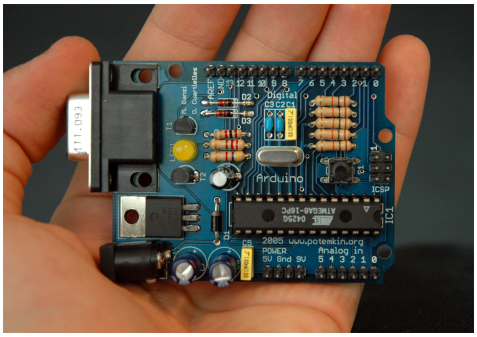
Figure 6.3: Comparison of the complexity between: the system already developed and the new proposed implementation. On the former case, each component needed soldering and careful handling. On the latter, components can be directly interfaced by cable or socket.

potential tool for water component detection and monitoring. They offer performances similar to commercial systems when derivative analysis (explained in section 6.2.2) is used to interpret the signal.

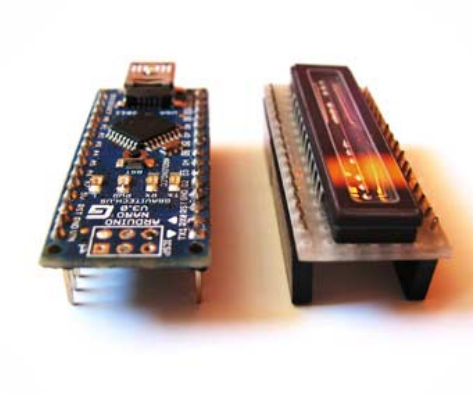
We propose to go a step further on lowering the cost and easing the implementation, making it more suitable to be developed by citizens. We propose to replace all the electronics from the cited system, except the microspectrometer, for an off-the-shelf solution like an Arduino board —not available at the time of the cited publication—. Arduino systems are open-source computer hardware microcontroller-based kits. These systems provide sets of digital and analog I/O pins that can be interfaced to various expansion boards and other circuits (figure 6.4). They are available commercially in preassembled form, or as do-it-yourself kits. The hardware design specifications are openly available, allowing the Arduino boards to be manufactured by anyone.

The spectrometer could be also replaced if desired; the Spectruino (mySpectral, 2015) is an open hardware Arduino based spectrometer. A *Spectruino DIY Kit* is being sold consisting of a Spectruino shield —shields are boards that can be plugged on top of the Arduino PCB extending its capabilities—, a diffraction grating, two optical slits and plans to build a Spectruino case from cardboard or polymethyl methacrylate (PMMA). Figure 6.3 exemplifies the simplicity of this approach.

Some known oceanographic projects using sensors coupled with Arduino hardware are: the do-it-yourself Remote Operated Vehicle (ROV) from Schneider (2011), where an Arduino Pro Mini microcontroller serves as the ROV brain and a second board as the data reader and display on the surface; the low-cost Autonomous Underwater Vehicle (AUV) from Busquets et al. (2012), where an Arduino Mega governs all the electronic elements; the spectrometer for water quality monitoring from Lakesh et al. (2014) using an Arduino with ATMEGA328P microcontroller; the underwater optical Sensorbot for in-situ pH monitoring from Johansen (2012), being an Arduino Pro Mini the core of the system; the multi-platform optical sensor for in-situ sensing of water



(a) Arduino board with an RS-232 serial, 14 digital I/O pins (upper/right) and 6 analog input pins (lower/right). Image from Nicholas Zambetti (CC BY-SA 3.0).



(b) Arduino Nano with Spectruino shield. With permission from Andrej Mosat ©2012 (mySpectral, 2015).

Figure 6.4: Arduino boards.

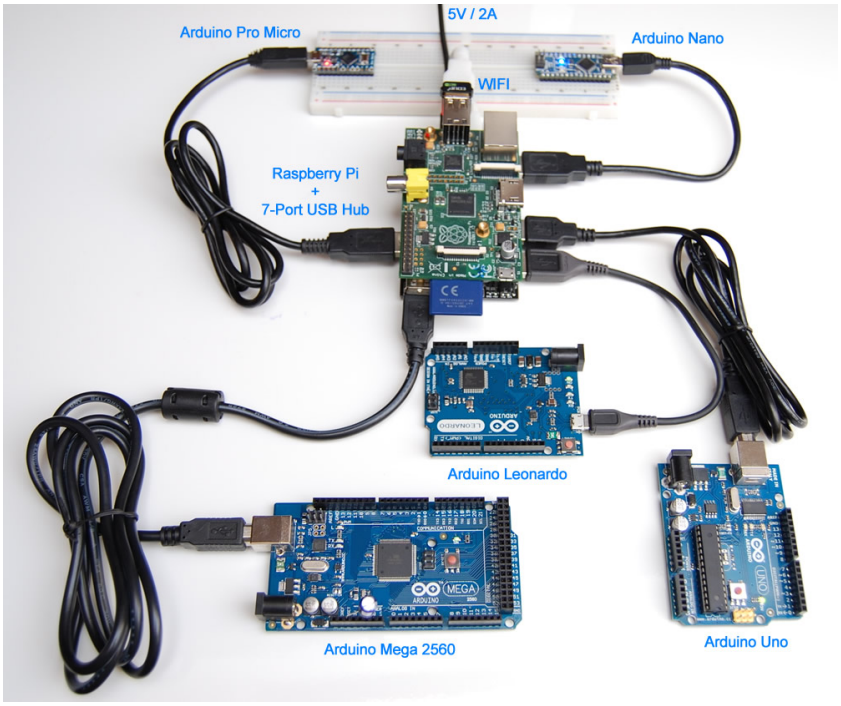


Figure 6.5: Raspberry Pi connected to five different Arduino boards with the help of a USB hub. With permission from UUGear.com ©2015 (UUGear, 2015).

Tree ID	Number of descriptors	Type of descriptors
a	11	Second derivative of K_d
b	13	K_d (m^{-1})
c	4	Concentration level (cell/L)

Table 6.1: Characteristics of the cases studied.

chemistry from Ng et al. (2012); and the low-cost moored buoy proposed by Wernand et al. (2012).

6.2.2 Data processing

Our objective is to detect if the concentration level of any of the harmful algal species from table 5.2 exceeds the threshold listed on this table (our definition of *bloom* according to section 5.2.2). As our time-series of K_d spectra has been generated from the data of chapter 5, we can associate each K_d spectrum to the goal predicate *bloom*.

Instead of using as descriptors the attributes corresponding to the 13 channels of the K_d spectrum, we calculate a new set of descriptors: the second derivative of the spectrum. Torrecilla Ribalta et al. (2012) demonstrated that the second derivative enhances shape singularities in hyperspectral data, which are significant since they are related to absorption features of phytoplankton pigments present in the samples. In the cited study, data clustering of spectral data from different algal groups performed better when using derived data than using non derived spectra. In our case, by deriving twice, we obtain 11 new descriptors for each spectrum. Sample derived K_d spectra are shown on figure 6.6.

These 11 descriptors are learned by a decision tree as in section 5.2.2. For comparison purposes, two additional decision trees are trained and test. One uses the original spectra, using the 13 channels measured by the sensor as descriptors. The other tree uses the concentration level of the 4 algal groups cited in section 6.2.1. The predicate goal is the same for the three systems: detect if the input descriptors correspond to a bloom event. A summary of the three systems is shown on table 6.1. All the systems have been implemented in Python using the Orange data-mining toolbox as in chapter 5, with the configuration described on page 39 . In order to avoid class imbalance, data has been undersampled as explained on page 41.

6.3 Results

Three different cases of prediction system have been studied (table 6.1), and the following indicators have been calculated: Accuracy, Precision, Sensitivity and Specificity. The prediction systems have been tested with a leave-one-out

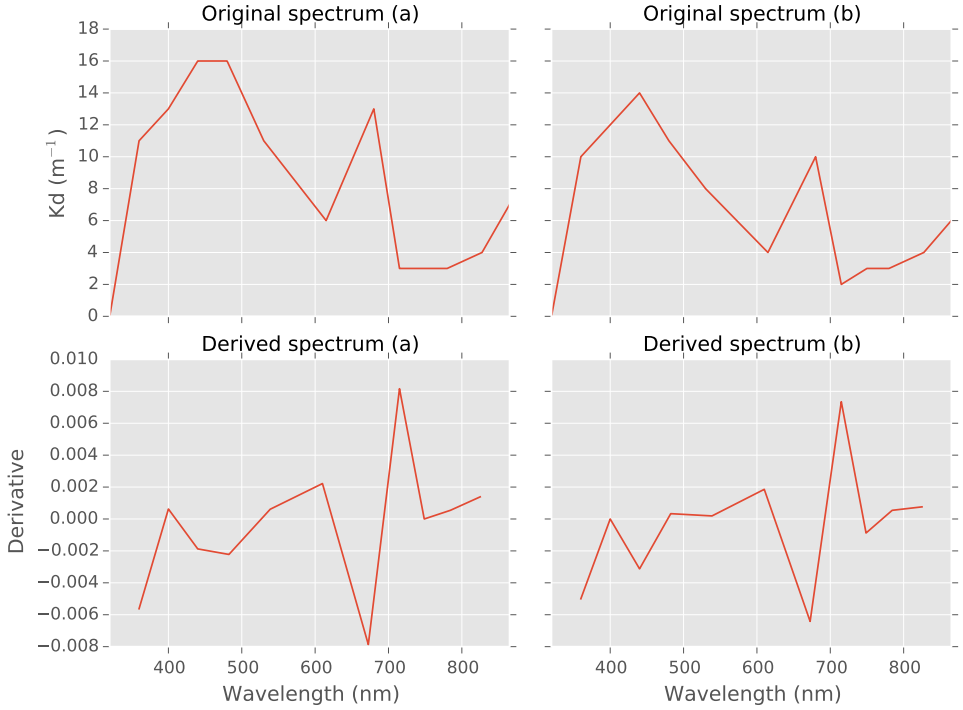


Figure 6.6: Top: Sample simulated K_d spectra for (a) a bloom event on 30th March, 2003 and (b) a non-bloom event on 13th June, 1999. Bottom: Second derivative of the former spectra.

Tree ID	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)
a	65	56	67	64
b	60	50	60	60
c	70	62	62	75

Table 6.2: Performance of the systems. Average data are shown. Performance is calculated using leave-one-out cross-validation.

cross-validation. See section 5.3 for a description of the indicators and the reasoning for choosing leave-one-out cross-validation. The results are shown in table 6.2.

6.4 Discussion

As in chapter 5, the objective of this study is to be able to detect the maximum number of bloom cases, minimizing at the same time the number of false negatives, so that the system can be trusted. The priority is not to miss

blooms, even at the cost of increasing false negatives. Therefore, the most relevant indicator, to evaluate the performance of the system, is the sensitivity, as it is (inversely) related to how many true blooms the system is missing. The second most relevant indicator is the precision, as it represents the proportion between true positives and false positives.

System *c* offers the best performance indicators of the three cases, except for having a sensitivity 5% lower than system *a*. The better performance of system *c* could be attributed to the fact that its input data (concentration of algal groups) is closer to the original data (concentration of algal species) used to set the bloom detection thresholds. Systems *a* and *b* use as input the output of a model which has been fed with system's *c* data.

In our experiment, the second derivative performs better than using the original spectra, according to previous studies (Torrecilla Ribalta et al., 2012).

6.5 Conclusions

The present research is a first step to demonstrate the feasibility of using low-cost and low-consumption hyperspectral sensors in combination with derivative spectroscopy to extract qualitative information of different water samples. It does not go in depth on testing different hyperspectral signal analysis techniques, neither mentions topics such as phytoplankton discrimination (Aymerich et al., 2014) or unmixing (Aymerich et al., 2010) that could be applicable to our scenario and that the authors are aware of. But this first analysis with decision trees based only on optical data provided a bloom alert system with and accuracy of 65% and a sensitivity of 67%, opening the doors to keep exploring this path.

The proposed processing was performed on a desktop computer, but it could run in-situ and in real-time. The Arduino board proposed on section 6.2.1 can be connected to the I/O ports of a Raspberry Pi (figure 6.5), similarly to chapter 4. The system could be connected to the shore by cable or radio frequency and send alarms in real time. The do-it-yourself philosophy behind all the components listed allows citizens to build their own hyperspectral sensors. This initiative could form part of citizen science projects focused on using optical data for water characterization such as Citclops (Wernand et al., 2012).

The use of both open hardware and open source allow users to not only collect data, but to experiment with the system and improve it, sharing their discoveries with the community. We believe that showing science as an activity where the general public can engage actively and discuss with other peers outside from the scientific community will blur the barrier between scientists and citizens. Citizens will benefit from understanding science better and enjoying participating in it, and scientists will benefit from crowdsourcing and their work being more recognized and appreciated.

Chapter 7

General conclusions

We started this thesis by stating some of the challenges that oceanography is facing. In the last few years, there has been an explosion in the amount of data that is available. But data alone are insufficient and need to be deciphered, analyzed, interpreted, and modeled. The traditional method of turning data into knowledge relies on manual analysis and interpretation. For long-term monitoring applications, this form of proceeding is inefficient and expensive. The information donated voluntarily by citizens can be used as a low-cost labor way to find solution to this problems. The oceans and coasts are in peril and we must use the full range of resources to effect positive change. Citizen science is one such tool that has been underutilized.

We believe that open source software coupled with low-cost do-it-yourself hardware can help to close the gap between science and citizens in the oceanographic field. This is the objective of this thesis, to demonstrate how open source software and low-cost hardware are effectively applied to oceanographic research and how can it develop into citizen science.

We continue analyzing four different scenarios where to demonstrate this idea.

The first case, chapter 3, is an example of using open source software for video analysis. This analysis is based on an existing study where video footage of adult lobster males were monitored and its motion quantified. We adapt ans implement known motion detection techniques, using open source libraries, to this scenario. Although according to the original publication the previous manual methodology was enough to detect the circadian rhythms, the proposed automatized methodology reveals that some information was lost. The new system shows not only similar results, but periodicities not found before.

We also propose that the motion detection algorithm could be complemented with ground truthing performed by volunteers. Problematic frames or segments where the automatic algorithm did not perform well could be delivered to volunteers (e.g., making them available online) that would easily

identify the real position of the lobster and provide feedback to the system. In addition to that, if the data were made public alongside the code, any citizen with software coding knowledge could try to implement new motion detection algorithms and compete between them for the most robust solution.

Next, chapter 4 builds over the previous chapter. It analyzes the benefits of using similar video processing techniques on footage from field recordings and processed in-situ, as a proof of concept of its feasibility. Real footage recorded by a submarine video-camera is analyzed on a Raspberry Pi simulating a submarine installation. The system is able to detect and classify specimens, using only open source libraries and off-the-shelf low-cost hardware already available to build such systems.

With the mentorship of scientists, citizens living close to shallow water masses could build and deploy their own amateur observational underwater nodes with a limited investment. Sharing the captured preprocessed footage with other users and classifying it interactively would be an unprecedented source of data for characterizing the ecosystem that would complement the current scientific and systematic approach.

Chapter 5 introduces open source machine learning software as a method to improve biological observations. In this case, using data from an existing monitoring field campaign of weekly sampling and microscopy measurements. We implement a forecast system that explores temporal patterns in these series of harmful phytoplankton species abundances. Decision trees are used to infer the probability of having a toxic Harmful Algal Bloom on the following one, two or three weeks, respectively. The system is able to achieve a 65–76% of success in precision, which quantifies how many bloom events the system is missing. Precision achieve rates of 80%–85%, which quantifies how many of the detected blooms are not false alarms.

As in chapter 3, if the data were made public alongside the code, users with software coding skills could try to improve the current forecasting algorithm and engage on an open discussion about machine learning applied to marine biology. Additionally, this project could be coupled with other initiatives focused on phytoplankton monitoring. Projects where volunteers help on the sampling process or the data analysis would benefit from being able to forecast when a bloom is going to happen, optimizing the mobilization of the volunteers.

The last case studied, chapter 6, is based on the findings of chapter 5. We present some preliminar results, as proof of concept, of how the manual water sampling and algal concentration measurements from the previous chapter could be replaced by low-cost do-it-yourself hardware with a miniaturized hyperspectral sensor. First we simulate the optical data that would be captured by a system of these characteristics. Next, this data is processed using a decision tree as before to estimate if the concentration from a harmful algal species exceeds an alarm threshold. On this first exploration we obtain an accuracy of 65% and a sensitivity of 67%, opening the doors to keep exploring

this path.

The chapter also analyzes how this project could be implemented with off-the-shelf low-cost components, allowing citizens to build their own hyperspectral sensors. As exposed in chapter 4, The use of both open hardware and open source allow users to experiment with the system and improve it, sharing their discoveries with the community, while generating innovative new sources of data.

To summarize, we showed how open source software and low-cost do-it-yourself hardware:

- offer better results than previous methodologies;
- use less resources (human and technical) than previous methodologies;
- allow to apply techniques already used in other fields to oceanography;
- open the doors to implicate volunteers on gathering new data by themselves and improve existing techniques.

The tools that we have developed and analyzed in this thesis should help stakeholders, policy-makers, educators, conservation practitioners, and researchers who wish to develop a marine or coastal citizen science program. We also hope that the insights and recommendations that came out of our discussions will stimulate further research on and assessment of marine and coastal citizen science programs.

It is very unlikely that citizen science will ever replace traditional marine monitoring efforts as some tasks are not amenable to volunteers (e.g., specialist equipment required, inaccessible locations, frequency of reporting), but it is clear that citizen science can play a large and increasingly important supplemental role in future evidence provision, science, and monitoring. The increasingly large spatial scales that are addressed by policy makers (e.g., regional, global) and the reduction of funding means that new methods are needed to provide the evidence-base. Citizen science is one method of addressing tasks that cannot easily be fully automated and providing data at scales that would not be possible using scientists alone.

Bibliography

J. Aguzzi, P. Abelló, et al. Locomotor activity rhythms of continental slope nephrops norvegicus (decapoda: Nephropidae). *Journal of Crustacean Biology*, 24(2):282–290, 2004.

J. Aguzzi, D. Sarriá, J. García, J. Del Rio, F. Sardà, and A. Manuel. A new tracking system for the measurement of diel locomotor rhythms in the Norway lobster, *Nephrops norvegicus* (L.). *Journal of Neuroscience Methods*, 173(2):215–224, 2008.

J. Aguzzi, C. Costa, Y. Fujiwara, R. Iwase, E. Ramirez-Llorda, and P. Menesatti. A Novel Morphometry-Based Protocol of Automated Video-Image Analysis for Species Recognition and Activity Rhythms Monitoring in Deep-Sea Fauna. *Sensors*, 9(11):8438–8455, 2009a.

J. Aguzzi, C. Costa, P. Menesatti, J. García, and F. Sardà. Monochromatic blue light entrains diel activity cycles in the norway lobster, *Nephrops norvegicus* (l.) as measured by automated video-image analysis. *Scientia Marina*, 2009b.

J. Aguzzi, C. Costa, K. Robert, M. Matabos, F. Antonucci, S. K. Juniper, and P. Menesatti. Automated image analysis for the detection of benthic crustaceans and bacterial mat coverage using the venus undersea cabled network. *Sensors*, 11(11):10534–10556, 2011.

I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications magazine, IEEE*, 40(8):102–114, 2002.

A. L. Alldredge and M. W. Silver. Characteristics, dynamics and significance of marine snow. *Progress in oceanography*, 20(1):41–82, 1988.

J. Allen, T. J. Smyth, J. R. Siddorn, and M. Holt. How well can we forecast high biomass algal bloom events in a eutrophic coastal sea? *Harmful Algae*, 8(1):70–76, 2008.

P. Andersen. *Design and implementation of some harmful algal monitoring systems*, volume 44. Unesco, 1996.

C. R. Anderson, R. M. Kudela, C. Benitez-Nelson, E. Sekula-Wood, C. T. Burrell, Y. Chao, G. Langlois, J. Goodman, and D. A. Siegel. Detecting toxic diatom blooms from ocean color and a regional ocean model. *Geophysical Research Letters*, 38(4), 2011.

D. M. Anderson, P. Andersen, V. M. Bricelj, J. J. Cullen, and J. J. Rensel. *Monitoring and management strategies for harmful algal blooms in coastal waters*. Unesco, 2001.

J. Au, P. Bagchi, B. Chen, R. Martinez, S. Dudley, and G. Sorger. Methodology for public monitoring of total coliforms, escherichia coli and toxicity in waterways by canadian high school students. *Journal of Environmental Management*, 58(3):213–230, 2000.

I. F. Aymerich, S. Pons, J. Piera, E. Torrecilla, and O. N. Ross. Comparing the use of hyperspectral irradiance reflectance and diffuse attenuation coeficient as indicators for algal presence in the water column. In *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2010 2nd Workshop on*, pages 1–4. IEEE, 2010.

I. F. Aymerich, A.-M. Sánchez, S. Pérez, and J. Piera. Analysis of discrimination techniques for low-cost narrow-band spectrofluorometers. *Sensors*, 15(1):611–634, 2014.

L. Backer, L. Fleming, A. Rowan, and D. Baden. Epidemiology and public health of human illnesses associated with harmful marine phytoplankton. *UNESCO Manual on Harmful Marine Algae. UNESCO/WHO*, pages 725–750, 2003.

C. R. Barnes, M. M. Best, and A. Zielinski. The neptune canada regional cabled ocean observatory. *Technology (Crayford, England)*, 50:3, 2008.

M. M. Best, P. Favali, L. Beranzoli, M. Cannat, M. N. Cagatay, J. J. Danobeitia, E. Delory, H. de Stigter, B. Ferre, M. Gillooly, et al. European multidisciplinary seafloor and water-column observatory (emso): Power and internet to european waters. In *Oceans-St. John's, 2014*, pages 1–7. IEEE, 2014.

K. Borne and Z. Team. The zooniverse: A framework for knowledge discovery from citizen science data. In *AGU Fall Meeting Abstracts*, volume 1, page 0650, 2011.

G. Bradski and A. Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc., 2008.

G. Bradski. The opencv library. *Doctor Dobbs Journal*, 25(11):120–126, 2000.

J. Busquets, J. V. Busquets, D. Tudela, F. Pérez, J. Busquets-Carbonell, A. Barberá, C. Rodríguez, A. J. García, and J. Gilabert. Low-cost auv based on arduino open source microcontroller board for oceanographic research applications in a collaborative long term deployment missions and suitable for combining with an usv as autonomous automatic recharging platform. In *Autonomous Underwater Vehicles (AUV), 2012 IEEE/OES*, pages 1–10. IEEE, 2012.

L. Campbell, R. J. Olson, H. M. Sosik, A. Abraham, D. W. Henrichs, C. J. Hyatt, and E. J. Buskey. First harmful dinophysis (dinophyceae, dinophysiales) bloom in the us is revealed by automated imaging flow cytometry. *Journal of Phycology*, 46(1):66–75, 2010.

O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Machine learning*, 46(1-3):131–159, 2002.

A. Chave, M. Arrott, C. Farcas, E. Farcas, I. Krueger, M. Meisinger, J. Orcutt, F. Vernon, C. Peach, O. Schofield, et al. Cyberinfrastructure for the us ocean observatories initiative: Enabling interactive observation in the ocean. In *Oceans 2009-Europe*, pages 1–10. IEEE, 2009.

N. V. Chawla, N. Japkowicz, and A. Kotcz. Editorial: special issue on learning from imbalanced data sets. *ACM Sigkdd Explorations Newsletter*, 6(1):1–6, 2004.

Q. Chen and A. E. Mynett. Predicting phaeocystis globosa bloom in dutch coastal waters by decision trees and nonlinear piecewise regression. *Ecological modelling*, 176(3):277–290, 2004.

Citizens' Observatory. Citizens' Observatory, 2015. URL http://www.citizen-obs.eu. [Online; accessed 22-October-2015].

E. S. Cochran, J. F. Lawrence, C. Christensen, and R. S. Jakka. The quake-catcher network: Citizen science expanding seismic horizons. *Seismological Research Letters*, 80(1):26–30, 2009.

C. C. Conrad and K. G. Hilchey. A review of citizen science and community-based environmental monitoring: issues and opportunities. *Environmental monitoring and assessment*, 176(1-4):273–291, 2011.

E. M. Conway. Drowning in data: Satellite oceanography and information overload in the earth sciences. *Historical studies in the physical and biological sciences*, 37(1):127–151, 2006.

J. Demšar, T. Curk, A. Erjavec, Črt Gorup, T. Hočevar, M. Milutinovič, M. Možina, M. Polajnar, M. Toplak, A. Starič, M. Štajdohar, L. Umek, L. Žagar, J. Žbontar, M. Žitnik, and B. Zupan. Orange: Data mining toolbox in python. *Journal of Machine Learning Research*, 14:2349–2353, 2013.

T. Desell, R. Bergman, K. Goehner, R. Marsh, R. VanderClute, and S. Ellis-Felege. Wildlife@home: Combining crowd sourcing and volunteer computing to analyze avian nesting video. In *2013 IEEE 9th International Conference on eScience*, pages 107–115. IEEE, 2013.

T. D. Dickey. Studies of coastal ocean dynamics and processes using emerging optical technologies. *Oceanography*, 17(2):09–10, 2004.

T. D. Dickey and R. R. Bidigare. Interdisciplinary oceanographic observations: the wave of the future. *Scientia Marina*, 69(S1):23–42, 2005.

G. Dubelaar and P. Gerritzen. Cytobuoy: a step forward towards using flow cytometry in operational oceanography. *Scientia Marina*, 64(2):255–265, 2000.

EyeOnWater. EyeOnWater, 2015. URL http://www.eyeonwater.org. [Online; accessed 18-October-2015].

U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37–54, 1996.

D. A. Fischer, M. E. Schwamb, K. Schawinski, C. Lintott, J. Brewer, M. Giguere, S. Lynn, M. Parrish, T. Sartori, R. Simpson, et al. Planet hunters: the first two planet candidates identified by the public using the kepler public archive data. *Monthly Notices of the Royal Astronomical Society*, 419(4):2900–2911, 2012.

S. R. Forrest, L. A. Coldren, S. C. Esener, D. B. Keck, F. J. Leonberger, G. R. Saxonhouse, and P. W. Shumate. Optoelectronics in japan and the united states. Panel report, Japanese Technology Evaluation Center, 1996.

C. Franzoni and H. Sauermann. Crowd science: The organization of scientific research in open collaborative projects. *Research Policy*, 43(1):1–20, 2014.

J. Gao-feng, T. Yong, and J. Yun-cheng. A service-oriented group awareness model and its implementation. In *Knowledge Science, Engineering and Management*, pages 139–150. Springer, 2006.

T. Gilbert. Maritime response operations-requirements for met/ocean data and services. In *Conference and workshop on meteorological and oceanographic services for marine pollution emergency response operations*, 1998.

C. Gini. *Variabilità e mutabilità*, volume 1. Ed. Pizetti E, Salvemini, T, 1912.

P. H. Gleick et al. *Water in crisis: a guide to the world's fresh water resources.* Oxford University Press, Inc., 1993.

N. Gordienko, O. Lodygensky, G. Fedak, and Y. Gordienko. Synergy of volunteer measurements and volunteer computing for effective data collecting, processing, simulating and analyzing on a worldwide scale. *arXiv preprint arXiv:1504.00806*, 2015.

G. Hines, A. Swanson, M. Kosmala, and C. Lintott. Aggregating user input in ecology citizen science projects. In *Twenty-Seventh IAAI Conference*, 2015.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, May-Jun 2007.

K. Hyder, B. Townhill, L. G. Anderson, J. Delany, and J. K. Pinnegar. Can citizen science contribute to the evidence-base that underpins marine policy? *Marine Policy*, 59:112–120, 2015.

M. Islam, J. McMullin, and Y. Tsui. Rapid and cheap prototyping of a microfluidic cell sorter. *Cytometry Part A*, 79(5):361–367, 2011.

R. Iwase, K. Asakawa, H. Mikada, T. Goto, K. Mitsuzawa, K. Kawaguchi, K. Hirata, and Y. Kaiho. Off hatsushima island observatory in sagami bay: Multidisciplinary long term observation at cold seepage site with underwater mateable connectors for future use. In *Scientific Use of Submarine Cables and Related Technologies, 2003. The 3rd International Workshop on*, pages 31–34. IEEE, 2003.

A. K. Jain. *Fundamentals of digital image processing.* Prentice-Hall, Inc., 1989.

N. Japkowicz. The class imbalance problem: Significance and strategies. In *Proc. of the Int'l Conf. on Artificial Intelligence.* Citeseer, 2000.

J. Johansen. *Underwater Optical Sensorbot for In Situ pH Monitoring.* PhD thesis, Arizona State University, 2012.

E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL http://www.scipy.org/.

K. Kim, T. Chalidabhongse, D. Harwood, and L. Davis. Real-time foreground–background segmentation using codebook model. *Real-Time Imaging*, 11(3): 172–185, 2005.

S. Kim, C. Robson, T. Zimmerman, J. Pierce, and E. M. Haber. Creek watch: pairing usefulness and usability for successful citizen science. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2125–2134. ACM, 2011.

V. Krebs. Motivations of cybervolunteers in an applied distributed computing environment: Malariacontrol. net as an example. *First Monday*, 15(2), 2010.

M. Kubat and S. Matwin. Addressing the curse of imbalanced datasets. In *One-sided Sampling Proceedings of the Fourteenth International Conference on Machine Learning. Nashville: Tennessee*, pages 178–186, 1997.

S. Lakesh, D. Dahanayaka, H. Tonooka, A. Minato, S. Ozawa, et al. Spectral signatures identifying instrument on spectroscopic measurement technique for water quality monitoring. In *Advanced Materials Research*, volume 838. Trans Tech Publ, 2014.

V. Lanfranchi, N. Ireson, U. When, S. Wrigley, and C. Fabio. Citizens' observatories for situation awareness in flooding. In *Proceedings of the 11th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2014): 18-21 May 2014*, pages 145–154, 2014.

J. Lee, M. Binkley, and R. Carlson. The great american secchi dip-in. *GIS World*, 10(8):42–44, 1997.

J. Lee, I. Hodgkiss, K. Wong, and I. Lam. Real time observations of coastal algal blooms by an early warning system. *Estuarine, Coastal and Shelf Science*, 65(1):172–190, 2005.

Z. Lee and K. L. Carder. Effect of spectral band numbers on the retrieval of water column and bottom properties from ocean color data. *Applied Optics*, 41(12):2191–2201, 2002.

J. W. Leis. *Digital signal processing using MATLAB for students and researchers*. John Wiley & Sons, 2011.

C. X. Ling and C. Li. Data mining for direct marketing: Problems and solutions. In *KDD*, volume 98, pages 73–79, 1998.

C. J. Lintott, K. Schawinski, A. Slosar, K. Land, S. Bamford, D. Thomas, M. J. Raddick, R. C. Nichol, A. Szalay, D. Andreescu, et al. Galaxy zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey. *Monthly Notices of the Royal Astronomical Society*, 389(3): 1179–1189, 2008.

S. B. Liu. Crisis crowdsourcing framework: designing strategic configurations of crowdsourcing for the emergency management domain. *Computer Supported Cooperative Work (CSCW)*, 23(4-6):389–443, 2014.

H. K. Lotze, M. Coll, A. M. Magera, C. Ward-Paige, and L. Airoldi. Recovery of marine animal populations and ecosystems. *Trends in Ecology & Evolution*, 26(11):595–605, 2011.

E. Louchard, R. Reid, C. Stephens, C. Davis, R. Leathers, T. Downes, and R. Maffione. Derivative analysis of absorption features in hyperspectral remote sensing data of carbonate sediments. *Optics Express*, 10(26):1573–1584, 2002.

H. R. Maier and G. C. Dandy. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental modelling & software*, 15(1):101–124, 2000.

J. Marra, C. Langdon, and C. A. Knudson. Primary production, water column changes, and the demise of a phaeocystis bloom at the marine light-mixed layers site (59 n, 21 w) in the northeast atlantic ocean. *Journal of Geophysical Research: Oceans (1978–2012)*, 100(C4):6633–6643, 1995.

A. Martin Cirera et al. Sistema de processat d'imatges per observatoris oceanogràfics. Master's thesis, Universitat Politècnica de Catalunya, 2010.

G. Massion and K. Raybould. Mars: The monterey accelerated research system-gene massion and keith raybould (monterey bay aquarium research institute) explore a cabled ocean observing system for a new generation of ocean. *Sea Technology*, 47(9):39–42, 2006.

S. Matuska, R. Hudec, and M. Benco. The comparison of cpu time consumption for image processing algorithm in matlab and opencv. In *ELEKTRO, 2012*, pages 75–78. IEEE, 2012.

C. R. McClain, K. Arrigo, K.-S. Tai, and D. Turk. Observations and simulations of physical and biological processes at ocean weather station p, 1951–1980. *Journal of Geophysical Research: Oceans (1978–2012)*, 101(C2): 3697–3713, 1996.

P. Menesatti, J. Aguzzi, C. Costa, J. García, and F. Sardà. A new morphometric implemented video-image analysis protocol for the study of social modulation in activity rhythms of marine organisms. *Journal of Neuroscience Methods*, 184(1):161–168, 2009.

T. M. Mitchell. *Machine learning. WCB*. McGraw-Hill Boston, MA:, 1997.

C. D. Mobley. *Light and water: Radiative transfer in natural waters*. Academic press, 1994.

C. D. Mobley and L. K. Sundman. *Hydrolight-Ecolight version 5.0 User's Guide*. Sequoia Scientific Inc, 2008.

S. Mockler. Water vapor in the climate system. *Special Report, American, Geophysical Union*, 1995.

N. Muttil and K.-w. Chau. Neural network and genetic programming for modelling coastal algal blooms. *International Journal of Environment and Pollution*, 28(3):223–238, 2006.

N. Muttil and J. H. Lee. Genetic programming for analysis and real-time prediction of coastal algal blooms. *Ecological Modelling*, 189(3):363–376, 2005.

mySpectral. mySpectral Arduino spectrophotometer: Spectruino, 2015. URL http://myspectral.com. [Online; accessed 17-October-2015].

National Economic Council and Office of Science and Technology Policy. *A strategy for American innovation*. White House, 2015.

U. Nations. *Water for people-water for life*. Unesco Publ., 2003.

E. Naylor. Rhythmic behaviour of decapod crustaceans. In *Symposia of the Zoological Society of London*, volume 59, pages 177–199, 1988.

E. Naylor. Chronobiology: implications for marine resource exploitation and management. *Scientia Marina*, 69(S1), 2005.

R. Neves and A. C. Matos. Raspberry pi based stereo vision for small size asvs. In *Oceans-San Diego, 2013*, pages 1–6. IEEE, 2013.

C.-L. Ng, S. Senft-Grupp, and H. F. Hemond. A multi-platform optical sensor for in situ sensing of water chemistry. *Limnology and Oceanography: Methods*, 10(12):978–990, 2012.

M. P. Nusbaum and M. P. Beenhakker. A small-systems approach to motor pattern generation. *Nature*, 417(6886):343–350, 2002.

S. Olenin, F. Alemany, A. Cardoso, S. Gollasch, P. Goulletquer, M. Lehtiniemi, T. McCollin, D. Minchin, L. Miossec, A. O. Ambrogi, et al. Marine strategy framework directive. Joint report, European Commission Joint Research Centre, 2010.

T. E. Oliphant. Python for scientific computing. *Computing in Science & Engineering*, 9(3):10–20, 2007.

J. Ortega-Escobar. Circadian rhythms of locomotor activity in lycosa tarentula (araneae, lycosidae) and the pathways of ocular entrainment. *Biological rhythm research*, 33(5):561–576, 2002.

R. Person, Y. Aoustin, J. Blandin, J. Marvaldi, and J. Rolin. From bottom landers to observatory networks. *Annals of geophysics*, 49(2/3):581–593, 2007.

M. Pocock, D. Chapman, L. Sheppard, and H. Roy. A strategic framework to support the implementation of citizen science for environmental monitoring. Final report, SEPA. Centre for Ecology & Hydrology, Wallingford, Oxfordshire, 2014.

S. Pons, L. Ceccaroni, and J. Piera. Design of a sensor network with adaptive sampling. In *iEMSs 2008 Conference (The International Environmental Modelling and Software Society Conference)*, 2008.

S. Pons, I. F. Aymerich, E. Torrecilla, and J. Piera. Monolithic spectrometer for environmental monitoring applications. In *OCEANS 2007-Europe*, pages 1–3. IEEE, 2007.

K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov. Real-time computer vision with opencv. *Communications of the ACM*, 55(6):61–69, 2012.

J. R. Quinlan. *C4. 5: programs for machine learning*, volume 1. Morgan Kaufmann, 1993.

R. Quinlan. Data mining tools see5 and c5.0, 2004.

F. Recknagel. Anna–artificial neural network model for predicting species abundance and succession of blue-green algae. *Hydrobiologia*, 349(1-3):47–57, 1997.

F. Recknagel, M. French, P. Harkonen, and K.-I. Yabunaka. Artificial neural network approach for modelling and prediction of algal blooms. *Ecological Modelling*, 96(1):11–28, 1997.

A. Rosenfeld and E. B. Troy. Visual texture analysis. Technical report, Maryland Univ., College Park (USA). Computer Science Center, 1970.

J. C. Russ. *The image processing handbook*. CRC press, 2011.

Y. Rzhanov, S. Pe'eri, and A. Saskov. Probabilistic reconstruction of color for species' classification underwater. In *OCEANS 2015-Genova*, pages 1–5. IEEE, 2015.

B. Schiele and J. L. Crowley. Object recognition using multidimensional receptive field histograms. In *Computer Vision—ECCV'96*, pages 610–619. Springer, 1996.

B. Schlining and N. J. Stout. Mbari's video annotation and reference system. In *OCEANS 2006*, pages 1–5. IEEE, 2006.

D. Schneider. Build your own robosub [hands on]. *Spectrum, IEEE*, 48(9): 24–26, 2011.

J. Serra. *Image analysis and mathematical morphology*. Academic Press, Inc., 1983.

A. Simon and P. Shanmugam. A new model for the vertical spectral diffuse attenuation coefficient of downwelling irradiance in turbid coastal waters: validation with in situ measurements. *Optics express*, 21(24):30082–30106, 2013.

C. Sivapragasam, N. Muttil, S. Muthukumar, and V. Arun. Prediction of algal blooms using genetic programming. *Marine pollution bulletin*, 60(10): 1849–1855, 2010.

A. R. Smith. Color gamut transform pairs. In *ACM Siggraph Computer Graphics*, volume 12, pages 12–19. ACM, 1978.

D. Solomatine, L. See, and R. Abrahart. Data-driven modelling: concepts, approaches and experiences. In *Practical hydroinformatics*, pages 17–30. Springer, 2008.

H. M. Sosik and R. J. Olson. Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. *Limnol. Oceanogr. Methods*, 5:204–216, 2007.

D. Spinellis and C. Szyperski. How is open source affecting software development? *Software, IEEE*, 21(1):28–33, 2004.

M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 111–147, 1974.

R. P. Stumpf, M. C. Tomlinson, J. A. Calkins, B. Kirkpatrick, K. Fisher, K. Nierenberg, R. Currier, and T. T. Wynne. Skill assessment for an operational algal bloom forecast system. *Journal of Marine Systems*, 76(1): 151–161, 2009.

S. Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1): 32–46, 1985.

M. J. Swain and D. H. Ballard. Color indexing. *International journal of computer vision*, 7(1):11–32, 1991.

E. Torrecilla, I. F. Aymerich, S. Pons, and J. Piera. Effect of spectral resolution in hyperspectral data analysis. In *Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International*, pages 910–913. IEEE, 2007.

E. Torrecilla, J. Piera, M. Vilaseca, et al. *Derivative analysis of hyperspectral oceanographic data*. INTECH Open Access Publisher, 2009.

E. Torrecilla Ribalta, J. Piera Fernàndez, and M. Vilaseca Ricart. *Novel approach to improve the assessment of biodiversity of phytoplankton communities based on hyperspectral data analysis*. PhD thesis, Universitat Politècnica de Catalunya, 2012.

K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 255–261. IEEE, 1999.

UUGear. UUGear Solution: Raspberry Pi + Arduino, 2015. URL http://www.uugear.com/uugear-rpi-arduino-solution. [Online; accessed 22-October-2015].

V. Vapnik and O. Chapelle. Bounds on error expectation for svm. *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, pages 261–280, 1999.

V. Vujovic and M. Maksimovic. Raspberry pi as a wireless sensor node: Performances and constraints. In *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on*, pages 1013–1018. IEEE, 2014.

B. Wei, N. Sugiura, and T. Maekawa. Use of artificial neural network in the prediction of algal blooms. *Water Research*, 35(8):2022–2028, 2001.

M. R. Wernand, L. Ceccaroni, J. Piera, O. Zielinski, et al. Crowdsourcing technologies for the monitoring of the colour, transparency and fluorescence of the sea. *Proceedings of Ocean Optics XXI, Glasgow, Scotland*, pages 8–12, 2012.

P. Whigham and F. Recknagel. Predicting chlorophyll-a in freshwater lakes by hybridising process-based models and genetic algorithms. *Ecological Modelling*, 146(1):243–251, 2001.

H. Wilson and F. Recknagel. Towards a generic artificial neural network model for dynamic predictions of algal abundance in freshwater lakes. *Ecological Modelling*, 146(1):69–84, 2001.

K. Wong, J. H. Lee, and P. J. Harrison. Forecasting of environmental risk maps of coastal algal blooms. *Harmful algae*, 8(3):407–420, 2009.

D. D. Woods, E. S. Patterson, and E. M. Roth. Can we ever escape from data overload? a cognitive systems diagnosis. *Cognition, Technology & Work*, 4 (1):22–36, 2002.

S. Wrigley. Low-cost watercourse sensing for flood management and citizen engagement. In *The University of Sheffield Engineering Symposium*. Sheffield, 2014.

H. Zhang, W. Hu, K. Gu, Q. Li, D. Zheng, and S. Zhai. An improved ecological model and software for short-term algal bloom forecasting. *Environmental Modelling & Software*, 48:152–162, 2013.

Appendix A

Code repository

All the code developed for this thesis is publicly available, under the GNU General Public License version 3, at the GitHub repository https://github.com/sponsfreixes/phd_thesis.

GitHub is a web-based Git repository hosting service. Git is a widely used distributed version control system for software development. Git was initially designed and developed by Linus Torvalds for Linux kernel development in 2005. Like the Linux kernel, Git is free software distributed under the terms of the GNU General Public License version 2. GitHub offers both plans for private repositories and free accounts, which are usually used to host open-source software projects.

Appendix B

Proceedings

This chapter includes a verbatim copy of the previous published conference proceedings relevant for this thesis.