Computer Vision for Marine Environmental Monitoring
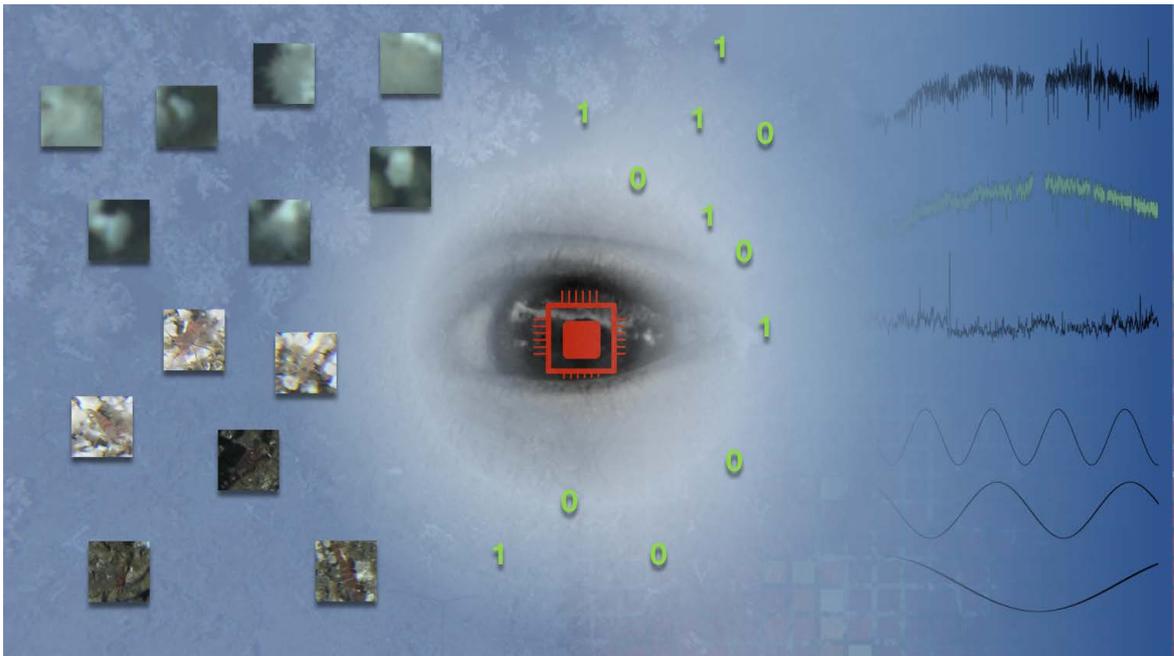

PhD Thesis

Jonas Osterloff
Bielefeld University
Faculty of Technology
Biodata Mining Group

Abstract

Ocean exploration using imaging techniques has recently become very popular as camera systems became affordable and technique developed further. Marine imaging provides a unique opportunity to monitor the marine environment. The visual exploration using images allows to explore the variety of fauna, flora and geological structures of the marine environment. This monitoring creates a bottleneck as a manual evaluation of the large amounts of underwater image data is very time consuming. Information encapsulated in the images need to be extracted so that they can be included in statistical analyzes. Objects of interest (OOI) have to be localized and identified in the recorded images. In order to overcome the bottleneck, computer vision (CV) is applied in this thesis to extract the image information (semi-) automatically. A pre-evaluation of the images by marking OOIs manually, i.e. the manual annotation process, is necessary to provide examples for the applied CV methods.

Five major challenges are identified in this thesis to apply of CV for marine environmental monitoring. The challenges can be grouped into challenges caused by underwater image acquisition and by the use of manual annotations for machine learning (ML). The image acquisition challenges are the optical properties challenge, e.g. a wavelength dependent attenuation underwater, and the dynamics of these properties, as different amount of matter in the water column affect colors and illumination in the images. The manual annotation challenges for applying ML for underwater images are, the low number of available manual annotations, the quality of the annotations in terms of correctness and reproducibility and the spatial uncertainty of them. The latter is caused by allowing a spatial uncertainty to speed up the manual annotation process e.g. using point annotations instead of fully outlining OOIs on a pixel level. The challenges are resolved individually in four different new CV approaches. The individual CV approaches allow to extract new biologically relevant information from time-series images recorded underwater.

Manual annotations provide the ground truth for the CV systems and therefore for the included ML. Placing annotations manually in underwater images is a challenging task. In order to assess the quality in terms of correctness and reproducibility a detailed quality assessment for manual annotations is presented. This includes the computation of a gold standard to increase the quality of the ground truth for the ML.

In the individually tailored CV systems, different ML algorithms are applied and adapted for marine environmental monitoring purposes. Applied ML algorithms cover a broad variety from unsupervised to supervised methods, including deep learning algorithms. Depending on the biologically motivated research question, systems are evaluated individually.

The first two CV systems are developed for the *in-situ* monitoring of the sessile species *Lophelia pertusa*. Visual information of the cold-water coral is extracted automatically from time-series images recorded by a fixed underwater observatory (FUO) located at 260 m depth and 22 km off the Norwegian coast. Color change of a cold water coral reef over time is quantified and the polyp activity of the imaged coral is estimated (semi-) automatically. The systems allow for the first time to document an *in-situ* change of color of a *Lophelia pertusa*

coral reef and to estimate the polyp activity for half a year with a temporal resolution of one hour.

The third CV system presented in this thesis allows to monitor the mobile species shrimp *in-situ*. Shrimp are semitransparent creating additional challenges for localization and identification in images using CV. Shrimp are localized and identified in time-series images recorded by the same FUO. Spatial distribution and temporal occurrence changes are observed by comparing two different time periods.

The last CV system presented in this thesis is developed to quantify the impact of sedimentation on calcareous algae samples in a *wet-lab* experiment. The size and color change of the imaged samples over time can be quantified using a consumer camera and a color reference plate placed in the field of view for each recorded image.

Extracting biologically relevant information from underwater images is only the first step for marine environmental monitoring. The extracted image information, like behavior or color change, needs to be related to other environmental parameters. Therefore, also data science methods are applied in this thesis to unveil some of the relations between individual species' information extracted semi- automatically from underwater images and other environmental parameters.

First, I would like to thank my supervisor Tim Nattkemper for his continues support and academic guidance. His positive motivation and believe in my skills allowed me to continue my PhD even in hard times when struggling with journal editors and reviewers. Furthermore, I would like to thank Timm for his support as a friend and college. It was always a pleasure to discuss and create new ideas for scientific research. He was the one who brought me in touch with marine informatics and even thinking to do a PhD. A very special thank to Ingunn. I really enjoyed the years working together and the many Skype phone calls. I also like to thank all my "underwater colleges" especially Torben, Daniel and Martin for the fruitful discussions and feedback during my time at the Bio Datamining Group.

I like to thank my family for their support over the years of doing my PhD and my grandfather who has inspired me to become a scientist. Last, but not least, Marina and our daughter Paula.

THANK YOU ALL!

Jonas Osterloff

Bielefeld, August 11, 2018

Contents

List of Figures

List of Tables

Acronyms

AC	annotation count
AQ	annotation quality
ACC	accuracy
ACE	automatic color equalization
ADCP	acoustic doppler current profiler
ANOVA	analysis of variance
AU	spatial annotation uncertainty
AUIP	automated underwater image preprocessing
AUV	autonomous underwater vehicle
BIIGLE	benthic image indexing, graphical labeling and exploration
BMU	best matching unit
CA	calcareous algae
CART	classification and regression trees
CCF	central composite face (design)
CFUO	camera equipped fixed underwater observatory
CH	calinski harabasz
CLAHE	contrast limited adaptive histogram equalization
CV	computer vision
DB	davies bouldin
DNN	deep neural network
DOPs	dynamics of the optical properties
ECC	enhanced correlation coefficient
FIA	full image annotation
FPR	false positive rate
FUO	fixed underwater observatory platform
FWB	fixed white balancing
GPGPU	general purpose graphical processing unit
H^{2}SOM	hierarchical hyperbolic self organizing map
HE	histogram equalization
HSI	hue, saturation, intensity color space
HSV	hue, saturation, value color space
ILSVRC	imagenet large scale visual recognition challenge
LBP	local binary pattern
LoVe	lofoten -vesterålen

LSACS	local space average color scaling
MANOVA	multivariate analysis of variance
ML	machine learning
MLP	multiple layer perceptron
OFOS	ocean floor observatories system
OOB	out-of-bag
OOI	object of interest
OP	optical properties
PA	polyp activity
PCA	principal component analysis
PLS	partial least squares
RF	random forest
RGB	red, green, blue color space
ROV	remotely operated vehicle
SAM	shrimp abundance map
SLIC	simple linear iterative clustering
SOM	self organizing map
SPC	single patch confirmation
TCCF	temporal color contrast features
TN	true negatives
TP	false negatives
TP	false positives
TP	true positives
UCC	unsupervised color correction
WB	white balancing
WIS	whole image screening
WPR	white patch retinex
WPRGW	white patch retinex and gray world combination

1 Introduction

1.1 Motivation

The planet we are living on, Earth, the blue planet is covered by 72% with the oceans [Kaiser, 2011]. The oceans drive our climate and weather, regulate our temperature, deliver freshwater, provide food, energy and mineral resources [Field et al., 2002]. The oceans have an average depth of 7 km [Field et al., 2002] and provide 99% of the space available for living on Earth [Barange, 2010; Kaiser, 2011]. Most of the places are very hard to reach and no one has ever explored them as most of the oceans habitat is nearly 4000 m deep [Kaiser, 2011]. The oceans with their multiple habitats are an extremely species rich environment. At the same time basic knowledge of marine organisms living in their natural environment is scarce. Humankind depends on the integrity of the oceans, but understanding the dynamics of this environment has only started. Exploring the oceans to reveal complex coherences of this environment can be crucial to preserve live on Earth.

Ocean exploration is typically carried out by collecting samples of species, measuring environmental parameters like temperature and salinity, and by observation of species in their natural environment. For the latter, using imaging techniques has received increasing attention, as camera systems have become affordable and technique developed further like battery capacity and high dynamic range cameras. Visual exploration using images allows to explore the variety of fauna, flora and geological structures of this "alien" underwater environment. Depending on the mission, sensors, including digital cameras, are mounted on different kinds of sensor platforms, like autonomous underwater vehicles (AUVs), remotely operated vehicles (ROVs) or fixed underwater observatory sensor platform (FUOs) for different kinds of explorations. Sensor platforms are optimized to maximize either spacial extent or temporal resolution or are a compromise between both. The choosing of a sensor platform therefore has to account the required temporal and spatial resolution for particular missions.

One special kind of ocean exploration is marine environmental monitoring. Ecological parameters need to be monitored over time to document the status of the environment and to identify sudden changes or trends. Changes and trends can be caused by anthropogenic activities, like oil drilling activities. In order to identify such impacts, natural variation of a parameter like species behavior have to be identified. It also has to be considered that the monitoring process itself can impact the monitored surrounding. Therefore, monitoring has to be planed with care regarding the minimization of this effect. Imaging provides an opportunity for environmental monitoring with limited impact in contrast for instance to using box cores, removing samples from the environment. Nevertheless, also imaging can have an impact due to light and noise emission. One parameter for environmental monitoring is species behavior, which can be investigated for instance as species occurrence or feeding activity of a species over time.

Fixed underwater observatories (FUOs) allow for environmental monitoring over long periods with a high temporal resolution. The spacial extent ($\approx$ 1 m^2 to 20 m^2) is very limited

compared to mobile platforms such as AUVs, but the temporal resolution combined with the duration of monitoring, which can be in the magnitudes of years, is one of its main advantages. Equipped with camera systems they allow to investigate change in color of a species (Section 6.1.2), to estimate variation in abundance and / or location of a species (Section 6.2), or to monitor the feeding activity of a species (Section 6.1.3), over time.

Visual exploration, including marine environment monitoring, is creating an enormous amount of digital photos, i.e. images, that need to be analyzed. Information in these images have to be extracted, so that it can be represented in data tables to apply statistical analyzes. The sheer amount of images demands for an automation of this process using Computer Vision.

1.2   Computer Vision for Underwater Environmental Monitoring

The humans got beaten when it comes to image classification [He et al., 2016], at least for the classification of images from the ImageNet 2012 [Russakovsky et al., 2015]. In [He et al., 2016] an advancement for training deep neural networks (DNNs) was introduced, being able to train and investigate deeper and wider network architectures. Their method allows to train DNNs achieving only a 4.95% top-5 test error (results were reported in February 2015), which is lower than the top-5 test error of 5.1% for humans reported in [Russakovsky et al., 2015]. Of course image classification is a very specific and limited task. A list of objects present in an image has to be produced [Russakovsky et al., 2015] and a correct classification is said to be achieved, if the list (top-5) contains the ground truth label of the classified image. The top-5 method is chosen, as images may display complex scenes and therefore contain several objects. Identifying and localizing multiple objects in images is a much more challenging task for Computer Vision (CV). In 2016 the best participant in ImageNet Large Scale Visual Recognition Challenge reached a mean average precision of 0.66 [UNC Vision Lab, 2016] and the best participant in 2017 a mean average precision of 0.73 [UNC Vision Lab, 2017]. The ImageNet 2012 contains 1000 object classes with 1,200,000 training, 50,000 thousand validation and 100,000 test images per class [Russakovsky et al., 2015].

In this thesis CV is applied to identify and localize multiple objects in time-series images recorded underwater. The limited knowledge of the "alien" environment makes CV a challenging task and no off-the-shelf solutions exists. For most underwater CV applications a training set similar to the ImageNet 2012 is simply not available. The number of training images is usually by magnitudes smaller. Therefore, usually the first step to apply CV to underwater images is to generate initial training images for the different classes of investigated objects or regions of interest. For the sake of compactness it will not be differentiated between objects of interest (OOIs) and regions of interest without any restriction. Expert knowledge is essential to generate such training images by marking OOIs in a subset of images, referred to as image annotation (Chapter 3). This is a very time demanding process and the number of annotations per OOI class in this thesis will still be very low compared to the ImageNet 2012 dataset. In Section 6.1 in total only 1110 annotations are available and the number of images per class is even smaller. Another reason for this limited number of training images of course is the low number of images available for certain species. Additionally, resolutions

of the OOIs can be quite low due to the costs of image recording (e.g. Section 6.2 with resolutions starting at about 20×20 pixels). It is always a trade-off between resolution and size of imaged area. Thus, the manual and the computational identification and localization in underwater images is a difficult task.

CV for marine environmental monitoring differs not only in number of available training data and pixel resolution. The task is different. Identification and localization is not enough, as visual features like color changes of the OOIs need to be investigated. But the computational analysis does not end here. Biologically relevant information of the OOIs have to be extracted to conduct further statistical analyses. Functional correlations with traditional sensors, like temperature or conductivity meters have to be unveiled in order to start to understand the dynamics of this environment (Chapter 8). Applying CV is therefore only one part in such data science approaches.

In this thesis, CV is applied in three *in-situ* underwater monitoring contexts and in one *wet-lab* context. The applied CV algorithms are in principle standard CV algorithms, but have been modified and extended for the requirements of the individual application. The biological research question is different for all applications. A most accurate segmentation is needed in Section 6.1.2 and 7.1, the identification and location in Section 6.2) and a mixture of both in Section 6.1.3. Thus, all applications demand for an individual tailored CV system.

Five major challenges have been identified in the context of this thesis making CV for underwater images a challenging task. The challenges can be grouped by image acquisition and manual annotation challenges and are presented in the following. All five challenges had to be considered in each CV application.

1.2.1 Image acquisition challenges

The image acquisition challenges, OP challenge and DOP challenge, summarize the effects of imaging underwater. Both challenges are affecting colors (Figure 1.1) of the imaged OOIs and are therefore most crucial for the quantification of color changes in Section 6.1.2 and 7.1.

OP challenge The optical properties (OPs) of the medium water differ from the medium air [Funk et al., 1972]. Thus, imaging underwater is different to imaging in air as the light transmission is different [Funk et al., 1972; Jaffe, 1990; Schettini et al., 2010]. Three main effects are influencing the transmission of light trough water: attenuation, back- and forward scatter. The attenuation in water is wavelength dependent, resulting in different absorption rates for different wavelength with lowest absorption in clear water for a wavelength of 480 nm. In general absorption rates are higher than in air resulting in significantly shorter ranges of visibility. The effect of the wavelength dependent attenuation is visible as color shifts and low range of visibility in the recorded images. The backscattering effect is caused by the reflection of light on particles and inhomogeneities in the water. The effect is mostly visible
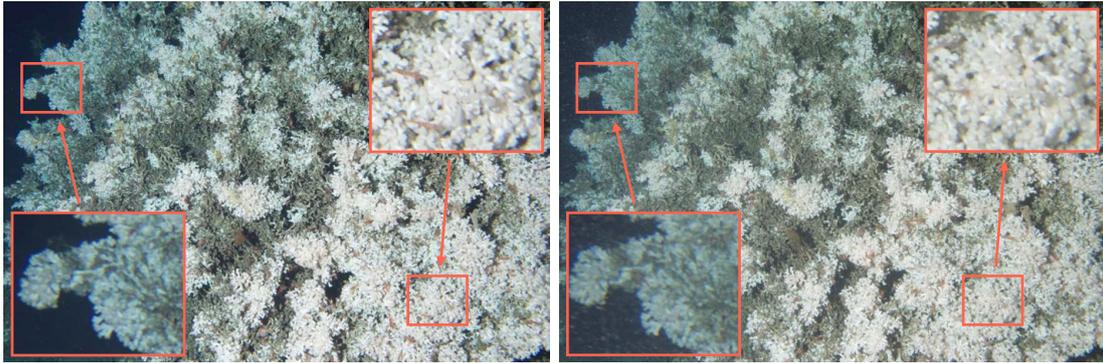
Figure 1.1: Effects of the image acquisition challenges. Both images were recorded by the LoVe observatory, but were not included in any further analysis in this thesis and feature a different camera orientation. Images were chosen as the effects of the image acquisition challenges becomes more clear. In each single image the effects of the OP challenge are visible. The camera-object distance influences the colors of the objects, so coral branches on the left which are located at a higher distance have a more blueish color than coral branches on the right which are located closer to the camera. Comparing both images, the effects of the DOP challenge are visible. Colors and image sharpness are affected due to the matter in the water column.

as a loss of contrast in the recorded images. The forward scattering is cause by the fraction on transparent plankton or differences of temperature or salinity concentrations. The effect is mostly visible as a blurring pattern in the recorded images. The intensity of all effects are dependent on the distance between the camera and imaged objects (camera-object distance) and on the distance between illuminant (e.g. a flash) and imaged object (illuminant-object distance). This distance dependency is mostly noticeable as color and illumination variations.

DOP challenge   The second image acquisition challenge are the dynamics of the optical properties (DOP). The optical properties of water can change dynamically as the inherent optical properties change with the amount of dissolved organic matter, particulate organic mater and particulate in-organic matter [Johnsen et al., 2009]. The change of matter in the water results in a change of the wavelength dependent attenuation. Attenuation rate is shifted towards longer wavelengths with increasing amount of dissolved organic material [Funk et al., 1972]. The colors of an image recorded during higher dissolved organic material concentration are shifted. Of course also image sharpness and contrast is influenced here. This change can occur quite rapidly for instance by a change of chlorophyll concentration as a result of an algae bloom event [Ingrid Kjerstad, 2016] and even a change in the concentration of salt is affecting the recorded images [Zhang et al., 2009].

1.2.2   Annotation challenges

Image examples for the different classes of OOIs are essential to be able to apply CV using machine learning (ML). Examples for each object/region class are essential to create CV

systems and to evaluate their performances. Therefore, OOIs need to be identified, classified and marked in a set of images by human observers. This is a challenging task especially for underwater images as the environment is different from the humans natural environment. Additionally, the previously described image acquisition challenges also affect the manual annotation (more details in Chapter 3).

AC challenge   The limited knowledge of the environment is also reflected in reduced manual annotation count (AC) in general and per class. The number of individuals of a class can be very low for rare classes, generating highly unbalanced training sets. For the selection of an ML algorithm this has to be considered. This can also lead to a change of the research question or a concentration on a class of OOIs with higher abundance.

AQ challenge   The annotation quality (AQ) is affected by the OP challenge and the DOP challenge challenge. Annotating underwater images manually is a challenging process regarding correctness and reproducibility. The correctness describes whether the annotation is correct, i.e. it has the correct class and correct location. The reproducibility describes the ability to repeat an annotation by re-performing the annotation process. Often only a limited number of expert biologists are able to identify OOIs, e.g. biota, visually in images and therefore annotate images with a high AQ. Annotations of a high quality are essential to create and evaluate CV systems as the manual annotations are representing the ground truth for the machine learning. This identification can be very time consuming and surprisingly error prone and therefore hard to reproduce. Visual identification on a 2D image is challenging in an alien environment. OOI are only visible from the side which they were recorded.

AU challenge   As mentioned earlier, annotating underwater images is a time demanding process. An OOI identification on a pixel level is often infeasible or too time-consuming. One strategy to speed up the annotation process is to over-segment or under-segment the OOI during the annotation process (see Figure 3.1). Thus a spatial annotation uncertainty (AU) is allowed. The allowed degree of AU is dependent on the biological research question and has to be considered in the evaluation of a CV system (Section 2.10).

1.2.3   Example

A short example is given here to illustrate the effect of all challenges. In CV OOIs are represented by feature vectors $\mathbf{v}^{(i)}$ which allow to group similar OOIs of the same class and differentiate between OOIs of different classes including a background class. The features $\mathbf{v}^{(i)}$ can represent a pixel, a region of connected pixels in an image or a whole image. To apply CV a minimal number of manual annotations including a class label $\omega^{(i)}$ is needed. The combination of $\mathbf{v}^{(i)}$ and $\omega^{(i)}$, both refer to the same image location, are used to train, validate and test a CV system.
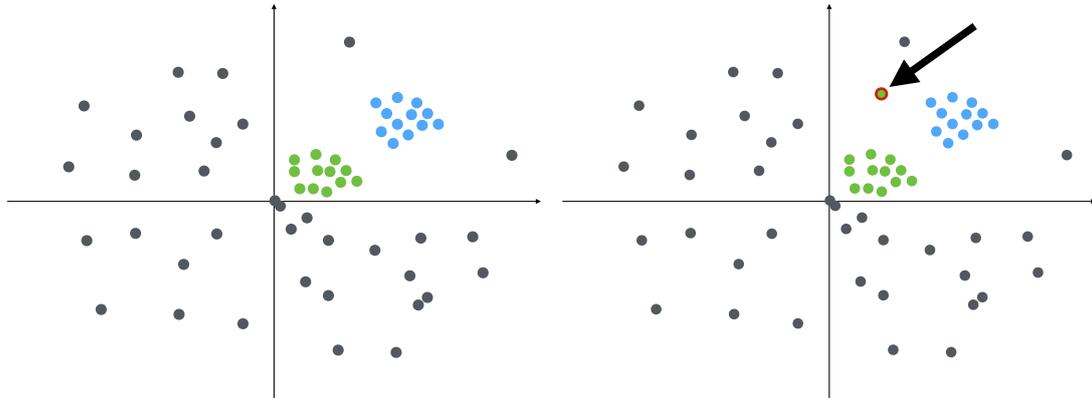
Figure 1.2: An artificial example of a feature space with two different classes of OOIs and one background class (black). The effects of the image acquisition challenges and the manual annotation challenges on a yet unclassified feature vector (arrow) are discussed in the text.

Having trained a CV system with a limited number of annotations, the general application is to classify new unseen feature vectors for instance from a new image $I_{t'}$. A new feature vector $\mathbf{v}^{(j)}$ may belong to a know class (green), but could be located in the feature space far from a group of features of the same class (Figure 1.2, green dot highlighted by an arrow). Coming back to the five challenges, this may have been caused by

1. the **OP challenge**: as the OOI is located at a different distance to the camera and appears in a different color, thus affecting the characteristics of $\mathbf{v}^{(j)}$

2. the **DOP challenge**: as the OPs have change and the OOI appears in a different color, thus affecting the characteristics of $\mathbf{v}^{(j)}$

3. the **AC challenge**: as the OOI is now imaged from a different and new side, which was not included in the training set

4. the **AQ challenge**: as the OOI is of a different class, but was annotated otherwise

5. the **AU challenge**: as the part of the OOI represented by the feature vector was not annotated

The combination of the underwater image acquisition challenges and the annotation challenges make CV for underwater imaging a challenging task. These five challenges will be addressed within this thesis. In the Sections 6, 7 the challenges will be solved individually for the different application.

1.2.4 Contributions

The six main contribution of this thesis are listed in the following:

- The guideline for manually annotating underwater images presented in Section 3. It is differentiated between full image annotation (FIA), whole image screening (WIS) and single patch confirmation (SPC). The guideline allows to increase the reliability and to assess the quality of manual annotations with respect to the different annotation tasks.

- The quality assessment for different underwater image preprocessing using cluster indices presented in Section 5.3. The use of manual annotations of different objects of interest allows to rank different preprocessing results for the discriminability in a chosen feature space. This optimization can be applied independently from the optimization the post applied machine learning.

- The Temporal Color Contrast Features (TCCF) presented in Section 6.2. TCCF allow to automatize the localization and identification of shrimp in underwater images despite their high variance in visual appearance due to the semi-transparent bodies and their variation and limitation in size. TCCF together with a stable camera-setup allows to filter for moving objects, boosting the performance of the automated localization and identification of shrimp instances.

- The color, illumination and zoom correction for a laboratory study presented in Section 7.1. The images reference plate is localized automatically and used to correct for color and zoom variation. It allows to quantify color and size changes of the investigated species.

- The color normalization presented in Section 6.1.2, allows to investigate long-term color changes of a cold-water coral reef using a camera with a stable orientation. The non-coral area is used to compensate temporary color shifts affecting the whole image. The normalization allowed for the first time to observe a color change of a cold-water coral reef *in-situ*.

- The holistic approach presented in Section 8.1 of integrating information extracted from images together with additional sensors to gain new insight into the biological behavior of a cold-water coral reef. Automatically extracted behavior information in form of feeding activity was for the first time analyzed for coherences with sensor data like current speed and direction, concentrations of chlorophyll, salt and temperature. A diurnal periodicity for both current and polyp activity was unveiled.

1.3 Publications

In the context of this thesis several manuscripts have been published. In the following the eight major publications are listed chronologically:

1. **Osterloff J**, Schoening T, Bergmann M, Durden JM, Ruhl HA, Nattkemper TW. **Ranking Color Correction Algorithms using Cluster Indices**. *2014 ICPR Workshop on Computer Vision for Analysis of Underwater Imagery,* Stockhom. [Osterloff et al., 2014b]
   Different image enhancement algorithms are ranked using Cluster Indices. In the field of underwater imaging it is often necessary to process images to homogenize image quality across a whole image transect and to compensate variations in imaging conditions. Four different methods were ranked regarding their performances on different image sets. The results show different rankings for four processing methods for two chosen sets benthic images from the deep sea.

2. **Osterloff J**, Nilssen I, Nattkemper TW. **A computer vision approach for monitoring the spatial and temporal shrimp distribution at the LoVe observatory.** *Methods in Oceanography.* 2016. [Osterloff et al., 2016a]
   Automated shrimp localization in time lapse images using supervised ML. Temporal color contrast features were applied to enable the detection of the semi-transparent shrimp. To study the spatialâĂŞtemporal characteristics of the shrimp, pseudo-color visualizations referred to as shrimp abundance maps (SAM) were introduced. SAMs for different time periods were presented, to show the potential of the methodology.

3. **Osterloff J**, Nilssen I, Eide I, de Oliveira Figueiredo MA, de Souza Tâmega FT, Nattkemper TW. **Computational Visual Stress Level Analysis of Calcareous Algae Exposed to Sedimentation.** *PLOS ONE.* 2016. [Osterloff et al., 2016c]
   Color and size of calcareous algae rhodolites were quantified semi-automatically using unsupervised ML. In a *wet-lab* study, calcareous algae rhodolites were exposed to different levels of sedimentation, light and water flow speed over 9 weeks. The experiment was conducted to investigate the potential impact of water-based drill cuttings on calcareous algae rhodolites. A customized CV system was developed to assess the temporal variation of the calcareous algae size ($\sim$ mass) and color automatically. Size and color were measured and were correlated to photosynthetic efficiency (maximum quantum yield of charge separation in photo-system II, $\Phi_{PSII_{max}}$) and degree of sediment coverage using multivariate regression. The multivariate regression showed correlations between time and calcareous algae sizes, as well as correlations between fluorescence and calcareous algae colors.

4. Schoening T, **Osterloff J**, Nattkemper TW. **RecoMIA - Recommendations for marine image annotation: Lessons learned and future directions.** *Frontiers in Marine Science.* 2016. [Schoening et al., 2016b]
   A guideline for the process of manually annotating underwater images. Furthermore, a statistical framework for the analyzes of the performance of human expert annotations

is given. This framework includes the computation of a ground truth in form of a gold standard.

5. **Osterloff J**, Nilssen I, Nattkemper TW. **Computational Coral Feature Monitoring for the fixed underwater Observatory LoVe.** *In: Proceedings of IEEE OCEANS 2016.* 2016. [Osterloff et al., 2016b]
   In-situ monitoring of color change of live cold water corals. Segmented cold water corals are analyzed for color change over more than half a year. Segmentation was achieved semi-automatically using unsupervised ML. Color normalizations allowed to differentiate between color changes of the whole image and color change of the coral area only. In this initial study, unexpected temporal change patterns could be revealed, by comparing color changes of live coral areas with color changes of coral rubble areas using the CIELab color space.

6. **Osterloff J**, Nilssen I, Järnegren J, Buhl-Mortensen P, Nattkemper TW. **Polyp activity estimation and monitoring for cold water corals with a deep learning approach.** *In: Proceedings of CVAUI 2016 (ICPR Workshop).* 2016. [Osterloff et al., 2016d]
   Deep learning was applied to estimate the activity of cold water coral polyps in time lapse images. Subsequent to a manual annotation study, polyp activity was estimated over half a year. Changes of the activity over this long-term period could be identified.

7. Durden J, Schoening T, Althaus F, Friedmann A, Garica R, Glover AG, Greinert J, Stout NJ, Jones DOB, Jordt A, Kaeli JW, Koser K , Kuhnz LA, Lindsay D, Morris KJ, Nattkemper TW, **Osterloff J**, Ruhl H, Singh H, Tran M, and Bett BJ. **Perspectives in Visual Imaging for Marine Biology and Ecology: From Acquisition to Understanding. Section: Image enhancement** *In: Hughes RN, Hughes DJ, Smith IP, Dale AC, eds. Oceanography and Marine Biology: An Annual Review. 54. Boca Raton: CRC Press;* 2016 [Durden et al., 2016b]
   In this Section of the journal paper, on overview of different preprocessing for underwater images is given. This overview allows to select a preprocessing suitable for a given camera orientation and illumination type (artificial vs natural) to correct for color, illumination and sharpness of the images.

8. **Osterloff J**, Nilssen I, Järnegren J, Van Engeland T, Buhl-Mortensen P, Nattkemper TW. **Computer vision enables short- and long-term analysis of Lophelia pertusa polyp behaviour and colour from an underwater observatory** *Scientific Reports* submitted July 2018 [Osterloff et al., submitted July 2018]
   In this journal paper, an initial approach is presented to investigate the functional relations of environmental sensor data, like temperature or salinity and species behavior in form of polyp activity for a period of half a year. The polyp activity of the cold water *Lophelia pertusa* was estimated automatically from images of an ocean observatory. A 24 hour rhythm of polyp activity could be identified which is likely correlated to changes in current speed. Furthermore, seasonal trends most likely caused by food supply were identified in polyp activity and color of the corals.

1.4   Structure

This thesis is structured as follows: In Chapter 2 the basics of Computer Vision in the context of the corresponding applications are introduced. In Chapter 3 the process of manually annotating underwater image is described and scores for annotation quality are defined. In Chapter 4 a brief introduction to underwater imaging is given, followed by a more detailed description of the images analyzed in this thesis. In Chapter 5 different image preprocessing methods are presented which can be applied to overcome the image acquisition challenges. The actual applications for CV are described in detail in Sections 6 and 7. First, in Chapter 6 CV is applied for *in-situ* monitoring of a cold water coral reef. Two different species are monitored, the sessile species *Lophelia pertusa* and the mobile species shrimp. Second, in Chapter 7 CV is applied in order to automatically analyze images recored in a *wet-lab* experiment. In these images the stress level of calcareous algae samples is quantified. In Chapter 8 the information extracted from images using CV are compared with other sensor data. In order to start to unveil some of the insights of the complex marine environment, data science is applied, integrating the inhomogeneous data in form of images and additional sensor data from a fixed underwater observatory. This data science approach is additionally applied to evaluate the *wet-lab* experiment. Functional relations between treatment variables and visual responses of the calcareous algae samples are investigated. In Chapter 9 individual solutions for the image acquisition challenges and annotation challenges in the different CV applications is discussed. Finally, in Chapter 10 the outcome of this thesis is summarized and a future perspective on the field of CV for environmental monitoring is given.

1.5   Notation

The notation in this thesis is organized as follows. Small letters in bold face refer to vectors (e.g. $\mathbf{v}^{(i)}$, $\mathbf{u}^{(k)}$), regular small letters refer to scalar values, including elements of a vector (e.g. $v_j^{(i)}$) and indices (e.g. j, i). Bold face capital letters refer to matrices including images (e.g. $\mathbf{I}$). Matrix elements are referred to as (e.g. $\mathbf{I}_t(x, y)$ or $\mathbf{I}_t(x, y, c)$). Other capital letters refer to sets or to amounts (e.g. N). In the following the most important symbols are listed with their corresponding meanings:

small Latin letters:

a	annotation
$a^{(i)}$	i-th annotation
c	color channel
$\mathbf{c}$	color vector
d	distance
f	sensor measurement
f_t	sensor measurement at time t

g	gold standard annotation
$g^{(j)}$	j-th gold standard annotation
h	pixel height of an image
i	index
j	index
k	index
o	observer
$o^{(i)}$	observer of the i-th annotation
q	quadrant of an image
r	radius
$\mathbf{r}$	a group of connected pixels, a region
$\mathbf{r}^{(i)}$	i-th region
t	time index
$\mathbf{u}$	prototype vector
$\mathbf{u}^{(k)}$	k-th prototype vector
$\mathbf{v}$	feature vector
$\mathbf{v}^{(i)}$	i-th feature vector
w	weight factor
w	pixel width of an image
x	horizontal position in an image
y	classification output
y	vertical position in an image
$y^{(i)}$	i-th classification output

capital Latin letters:

$\mathbf{B}$	binary image
C	set of channels
G	set of gold standard annotations
$\mathbf{I}$	an image
$\mathbf{I}_t$	an image at time point t
$\tilde{\mathbf{I}}$	preprocessed image for coral color quantification
$\tilde{\mathbf{I}}'$	preprocessed image for polyp activity estimation
$\breve{\mathbf{I}}$	preprocessed image for calcareous algae investigation
$\hat{\mathbf{I}}$	preprocessed image for shrimp identification and localization
K	amount of clusters
$\mathbf{M}$	image mask

N amount of something

O set of observers

S amount of cross validations

T time period

$\mathbf{W}$ weight matrix

small Greek letters:

α relative shrimp abundance

$\delta_{i,j}$ Kronecker delta

η pixel neighborhood in an image

θ polyp activity

θ_t polyp activity at time t

ν interpolated sensor measurement

ν_t interpolated sensor measurement at time t

ξ coral color feature

ξ_t coral color feature at time t

ρ image patch

ϕ morphological operator

ψ cluster index score

ω annotation class

$\omega^{(i)}$ i-th annotation class

capital Greek letters:

Δ classification function

Θ subset of feature vectors

Ψ image transformation function

Ω Set of annotation classes

The notation | | has several meanings depending on the enclosed type of symbol. For scalars this refers to the absolute value, for vectors it refers to the length of the vector, for sets it refers to the amount of elements in the set and for a connected pixel region $\mathbf{r}$ it refers number of pixels in the region.

2 Computer Vision

Computer Vision (CV) aims to automate the understanding of visual information [Gevers et al., 2012], to understand images [Sonka et al., 2008]. Images are typically a two-dimensional (2D) representation of a three-dimensional (3D) world. High level information needs to be extracted from these 2D images. This can for instance be the number of occurrences, locations, sizes or visual appearance such as color or feeding activity of specific objects of interest (OOIs) in images. Depending on the corresponding research question, OOIs either need to be located, classified, segmented or just inspected or combinations of those.

In order to fulfill a particular task, different algorithms need to be adapted and combined to a problem-specific CV system. In principle, CV consists therefore of several processing steps including image processing (Chapter 5), feature extraction (Section 2.3), machine learning (Section 2.7), post processing (Section 2.9) and evaluation (Section 2.10). The individual steps need to be adapted and tuned (Section 2.5, Section 2.8) to generate a solution with a sufficient performance, both in terms of time and quality.

Underwater CV has to deal with the image acquisition challenges (OP challenge and DOP challenge) and the annotation challenges (AQ challenge, AC challenge and the AU challenge). The effects of the image acquisition challenges are often reduced by a preprocessing (Chapter 5) nevertheless affect the subsequent processes as a fully compensation is often not reachable in the preprocessing. The annotation challenges are affecting several processes in a problem-specific CV pipeline and make the combination and selection of appropriate CV processes a challenging task. Furthermore, underwater images often feature a low resolution of the OOIs as for instance in Section 6.1.3, where the size of the OOIs is not bigger than 46×46 pixels. OOIs can also be semitransparent (Section 6.2), inducing a higher variation in the visual appearance. Hence, typical large scale object detection [Russakovsky et al., 2013] methods can not be applied at least not without modifications.

Individual methods have been chosen to tailor a problem-specific solution. In this Section all applied methods are introduced and the actual applications are presented in Chapter 6 and 7.

2.1 Digital image

A digital image $\mathbf{I}_t$ with its pixel values $\mathbf{I}_t(x, y, c)$ is a multi dimensional matrix of width w and height h and channels $c \in C$. Images in this thesis typically have three channels $C = \{r^{(\mathrm{RGB})}, g^{(\mathrm{RGB})}, b^{(\mathrm{RGB})}\}$ with intensities corresponding to the camera sensors with red, green and blue color filters. Intensities are either 8 bit $(0, .., 255)$ or 16 bit $(0, .., 65535)$. w and h are camera and camera-setting dependent and can vary for each project. In this thesis, images are recorded in time-series and camera settings are constant for each analyzed time-series of images.
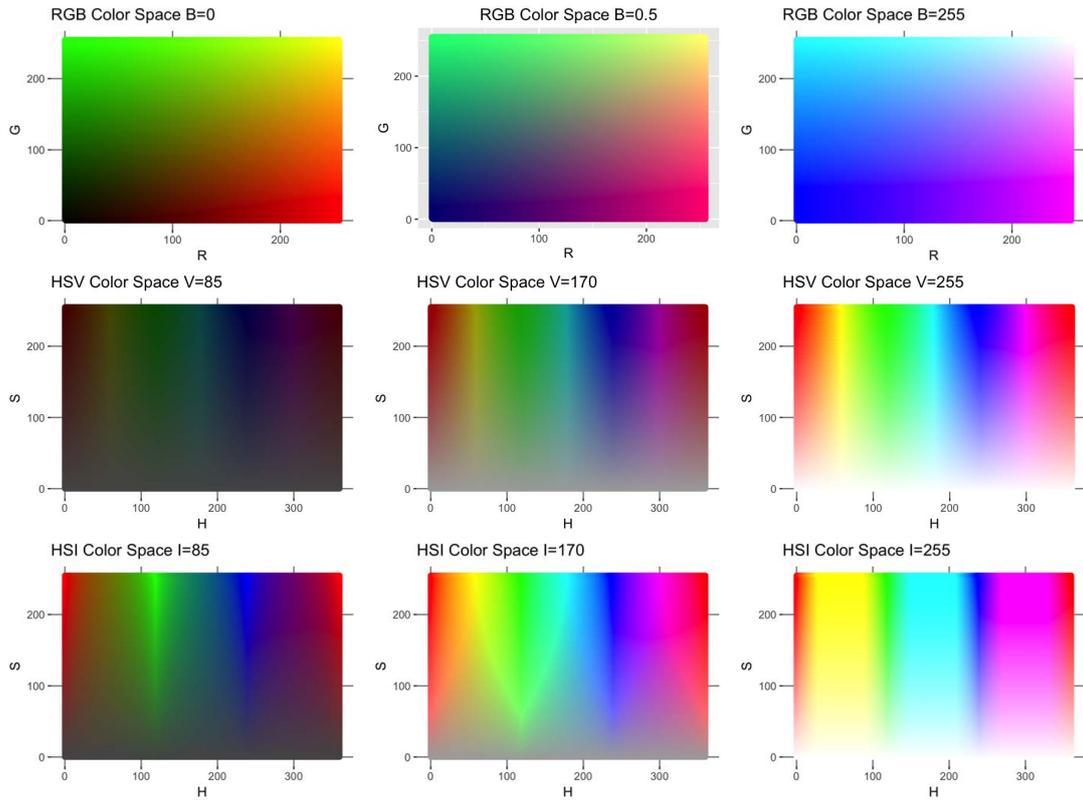
Figure 2.1: The 3D color spaces RGB, HSV and HSI. The RGB color space (upper row) is presented for $b^{(\text{RGB})} = 0$ (left column), $b^{(\text{RGB})} = 128$ (middle column) and $b^{(\text{RGB})} = 255$ (right column). The HSV color space (middle row) is presented for $v^{(\text{HSV})} = 85$ (left column), $v^{(\text{HSV})} = 170$ (middle column) and $v^{(\text{HSV})} = 255$ (right column). The HSI color space (middle row) is presented for $i^{(\text{HSI})} = 85$ (left column), $i^{(\text{HSI})} = 170$ (middle column) and $i^{(\text{HSI})} = 255$ (right column).

2.2   Color spaces

Images can be represented in different color spaces with different characteristics. Images can be transformed between color spaces for feature extraction (Section 2.3) and preprocessing of the images (Chapter 5), the latter for instance to increase color constancy. A selection of applied color spaces, visualized in Figure 2.1 and 2.2, is presented in the following.

2.2.1   RGB color space

The RBG color space is the standard color space of digital images. The three channels $r^{(\text{RGB})}, g^{(\text{RGB})}, b^{(\text{RGB})}$ specify the intensities for red, green and blue components, used for displaying on a screen [Ebner, 2007]. Depending on the color depth of the images $r^{(\text{RGB})}, g^{(\text{RGB})}, b^{(\text{RGB})}$ range either from 0 to 255 or 0 to 65536 (Section 2.1).
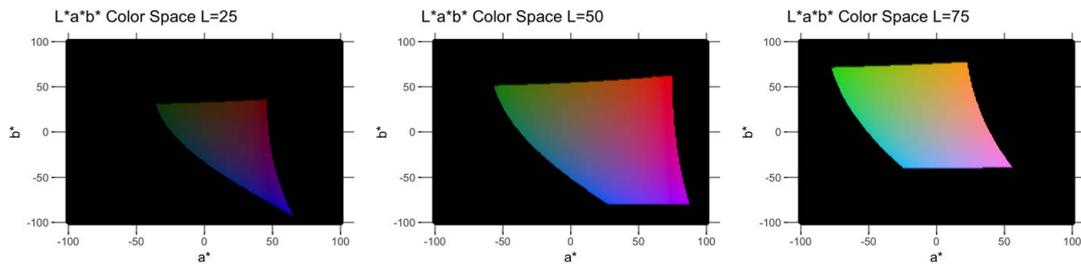
Figure 2.2: The L*a*b* color space. Due to compatibility reasons colors are transformed to RGB values for printing. This is therefore only a reduced L*a*b* color space, as not all L*a*b* colors are displayable in RGB.

2.2.2 CIE L*a*b* and CIE L*u*v color space

The CIE $L^*a^*b^*$ color space and the CIE $L^*u^*v^*$ color space are both perceptually uniform color spaces. Thus, color differences can be computed using the Euclidean distance, simulating color distances similar to the perceptually observed color differences [Ebner, 2007]. In both color spaces the intensity or luminance L^* is described independently of a color. In the $L^*u^*v^*$ space the u^* channel ranges from green over yellow to red and the v^* describes the blueness, where low values correspond to more blue. In the $L^*a^*b^*$ space a^* describes the greenness - redness, where higher values correspond to more redness, and b^* describes the blueness - yellowness, where higher values correspond to more yellowness [Ebner, 2007].

2.2.3 HSV color space

The HSV (hue, saturation, value) color space [Smith, 1978] is based on a cylindrical-coordinate system. The hue h^{HSV} is encoding the color. It starts at red ($h^{\mathsf{HSV}} = 0$) continues with yellow ($h^{\mathsf{HSV}} = 60$), green ($h^{\mathsf{HSV}} = 120$), blue ($h^{\mathsf{HSV}} = 240$) and ends again at red ($h^{\mathsf{HSV}} = 360$). The saturation s^{HSV} encodes the saturation of the color ranging from white/gray/black, depending on the value v^{HSV}, to full saturation. The value v^{HSV} encodes the brightness ranging from black to full brightness. Formulas for transformations between RGB and HSV color space can be found in[Ebner, 2007].

2.2.4 HSI color space

The HSI color space is very similar to the HSV color space. The color is again encoded as a hue h^{HSI}, but computed slightly different than h^{HSV}. Saturation s^{HSI} and intensity i^{HSI} differ from s^{HSV} and v^{HSV} as they are computed differently (Fig 2.1). Formulas for transformation between RGB and HSI color space can for instance be found in [Gonzalez et al., 2008].

2.2.5 $l\alpha\beta$ **color space**

The $l\alpha\beta$ color space, also know as the Ruderman component color space is a logarithmic color space [Ruderman et al., 1998]. The motivation of this color space is to have a color space with minimized redundancy between the different channels for natural images [Ruderman et al., 1998; Reinhard et al., 2001]. Similar to the $L^*a^*b^*$ color space, the l-channel represents the luminance and the α- and β-channels encode blue-yellow and red-green opponents. Formulas for transformation between RGB and HSI color space can be found in [Reinhard et al., 2001]. A useful feature of this color space is the simplification of white-balancing (Section 5.2.1). In this space the average of channels $\bar{\alpha}$ and $\bar{\beta}$ only need to be shifted to 0 for white-balancing [Bianco et al., 2015].

2.2.6 $Y'C_BC_R$ **color space**

The $Y'C_BC_R$ color space [Ebner, 2007; Poynton, 2012], also known as the $YCrCb$ color space, was originally invented for television broadcasting. It is a perceptually uniform space consisting of a luma channel and two chroma channels. The luma channel (Y') is similar to a luminance channel, but computed on the gamma transformed ($\gamma = 0.4$) RGB channels. C_R is a chroma channel computed from the scaled difference of the $r^{\text{(RGB)}}$ transformed ($r'^{\text{(RGB)}}$) and the luma channel. The C_B, also a chroma channel, is computed from the scaled difference of the transformed $b^{\text{(RGB)}}$ channel ($b'^{\text{(RGB)}}$) and the luma channel.

2.3   Feature extraction

In order to apply Machine Learning (ML) image features need to be extracted. A feature can be the distribution of colors in an image. The mathematical description of an image feature is given by its feature descriptor and applying a feature descriptor to an image is called feature extraction. Features can be computed on full images and on image regions, representing local image features. The latter is the typical type of feature extraction in this thesis. A region $\mathbf{r}^{(i)}$, a group of connected pixels, can either have a fixed shape (e.g. rectangles or circles) or a dynamic shape defined by local image similarities, e.g. by computing super-pixels (see Paragraph: Super-pixel). Feature vectors $\mathbf{v}^{(i)}$ are extracted on each region $\mathbf{r}^{(i)}$ by applying the feature descriptor to $\mathbf{r}^{(i)}$. The location of each $\mathbf{v}^{(i)}$ is given by the corresponding $\mathbf{r}^{(i)}$ localized in the image of extraction. In contrast to feature extractions with dynamic shapes, feature extractions with fixed shapes can be applied allowing an overlap between different $\mathbf{r}^{(i)}$. For simplification purposes therefore for fixed shapes the center, e.g. the center of mass, of the $\mathbf{r}^{(i)}$ is used as the location of resulting feature vector $\mathbf{v}^{(i)}$. The degree of overlap for fixed-shaped regions is modified by the step width s. It defines the minimal distance between center points of extractions in number of pixels in x and y directions (see Figure 2.3). For larger s this can also result in no overlap or even gaps between the extraction. Feature descriptors can be combined by concatenating corresponding feature vectors extracted on the same image $\mathbf{I}_t$ and region $\mathbf{r}^{(i)}$.

(a) fixed-shaped regions ($|\mathbf{r}| = 3 \times 3$)

(b) dynamic shaped regions
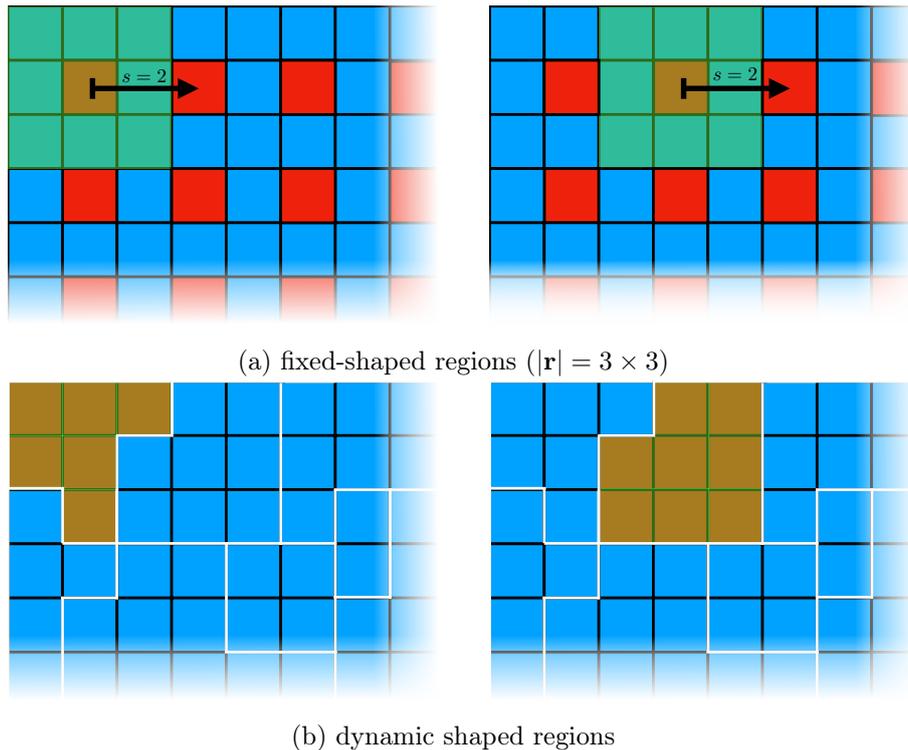
Figure 2.3: Feature extractions for regions with fixed shapes, here square (a) and regions with dynamic shapes (b). For the feature extraction with fixed shapes (a), image locations in form of the center are highlighted in red. Features are extracted with a step width $s = 2$ and therefore for every second pixel in x and y direction. For feature extraction with dynamic shapes (b), region boarders are highlighted in white. The location for the extracted feature is defined by the region $\mathbf{r}^{(i)}$ of extraction itself. For each region $\mathbf{r}^{(i)}$ one feature vector $\mathbf{v}^{(i)}$ will be extracted.

In the following features are described which were applied successfully within the scope of this thesis.

2.3.1   Local statistics

One of the simplest features are statistical representations of pixel intensities. The feature descriptor concatenates the pixel values $\mathbf{I}(x, y, c)$ with $(x, y) \in \mathbf{r}^{(i)}$. If regions $\{r^{(i)}\}$ contain only one pixel $\mathbf{I}(x, y, c)$, the feature vector is computed as:

$$\mathbf{v}^{(i)} = (I(x, y, 0), ..., I(x, y, |C| - 1))^T \, .$$

For example for an RBG image $\mathbf{I}$, feature vectors are computed as

$$\mathbf{v}^{(i)} = (I(x, y, \mathsf{red}), I(x, y, \mathsf{green}), I(x, y, \mathsf{blue}))^T \, .$$

Another approach is to compute statistical functions η on pixel values in $\mathbf{r}^{(i)}$ ($|\mathbf{r}^{(i)}| > 1$), where η is typically computed for each channel individually. For a region $\mathbf{r}^{(i)}$ in an RGB image for instance defined as:

$$\mathbf{v}^{(i)} = \begin{pmatrix} \eta\left(I(x, y, c = R)\right)_{(x,y)\in\mathbf{r}^{(i)}} \\ \eta\left(I(x, y, c = G)\right)_{(x,y)\in\mathbf{r}^{(i)}} \\ \eta\left(I(x, y, c = B)\right)_{(x,y)\in\mathbf{r}^{(i)}} \end{pmatrix}$$

η can be any channel-wise statistical measurement including the average, median, minimum or maximum of a region.

2.3.2   Histogram features

The frequency of pixel values in $\mathbf{r}^{(i)}$ is considered in the histogram descriptor. Frequencies are typically computed in intensity intervals, the intensity range is divided, which is called binning. Common numbers of intervals b are 4, 8 or 16 per color channel, if applied to image regions. The number of bins determines the dimension of the descriptor output $\dim(\mathbf{v}^{(i)}) = b \cdot |C|$. Applying the histogram feature descriptor to regions of varying sizes requires normalization by the regions size $|\mathbf{r}^{(i)}|$: $\frac{\mathbf{v}^{(i)}}{|\mathbf{r}^{(i)}|}$. Histogram descriptors can be extracted using different color spaces (see Section 2.2) or in combination with other descriptors of which output can be interpreted as another image, such as responses of 2D Gabor wavelets, local binary pattern (LBP) [Pietikäinen et al., 2011] or other edge descriptive responses.

2.3.3   Canny edge features

A feature can also be the information about the location of edges in an image. A very popular method for edge detection is the Canny edge filtering [Canny, 1986]. The algorithm can be
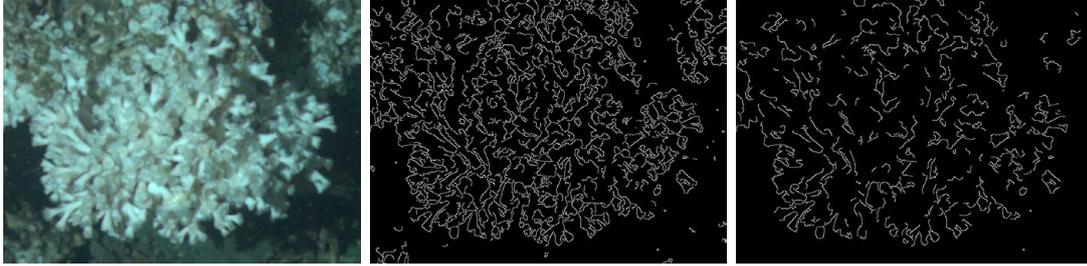
Figure 2.4: Canny edge filtering. The original image (left) and corresponding Canny edge responses for 2 different sets of thresholds (center = $40, 120$, right = $60, 180$) are shown. The influence of the double thresholding can be observed comparing the center and right image.

divided into four steps. First, a Gauss filtering is applied to smooth the image. Second, gradient magnitude, i.e. a convolution with a Sobel operator, and angle images (atan2) are computed, followed by applying of non-maxima suppression to the gradient magnitude image. Finally, double thresholding in form of a hysteresis thresholding and analysis of connectivity for identified edges is applied [Gonzalez et al., 2008]. This detection method allows to detect edges while ignoring irrelevant details of images and has thus become the standard method for edge detection [Gonzalez et al., 2008]. Results for applying the Canny edge detection form the output of the Canny edge descriptor and are shown in Figure 2.4. Canny edge detection can also be applied for image alignment (Section 6.2.2) and has shown to be a robust edge detection algorithm for underwater images.

2.3.4 Gabor features

Similar to the Canny edge descriptor, the Gabor descriptor uses the response images of a filtering process for the feature extraction. The idea of 2D Gabor filters is to model the visual cortex [Tai Sing Lee, 1996]. A Gabor filter consists of two elementary functions, a 2D Gaussian function and a complex plane wave function [Tai Sing Lee, 1996]. Typically multiple responses of Gabor filters with different scales and orientations of the wavelet function are used. As in [Ontrup et al., 2004], Gabor filters are computed using the following formula:

$$G(x,y) = e^{-\left(\frac{(x-x_o))^2}{2\sigma_x{}^2} + \frac{(y-y_o))^2}{2\sigma_y{}^2}\right)} e^{-ik(x-x_0)}$$

with (x_0, y_0) being the center of the filter (spatial domain), σ_x and σ_y being the width of the Gaussian part in x and y direction of the filter and k as the spatial frequency of the complex wave function along the x-axis. As in [Ontrup et al., 2004], five orientations and three scales of Gabor filters F_g ($g = \{1, ..., 5\}$) are used in this thesis to form a filter bank of Gabor filters (Fig 2.5). The Gabor descriptor, extracts features on the filter response images (example response image can be found in Figure 2.6) for instance using a histogram feature extraction (Section 2.3.2). Features are extracted on all response images and resulting $\mathbf{v}^{(i)}$ are concatenated.
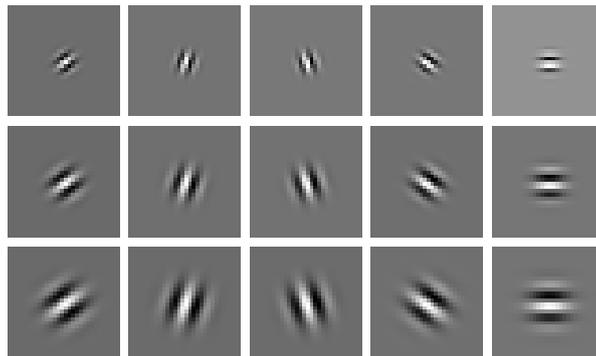
Figure 2.5: A Gabor filter bank with 3 scales and 5 orientations. Responses of the filtering are visualized in Figure 2.6.

2.3.5 Data-driven features

Data-driven features have become very popular, due to the increase of the accuracy achieved by Convolutional Neural Networks (CNN) (Section 2.7.3). Another method used to compute data-driven features is the Principal Component Analysis (PCA) [Gonzalez et al., 2008]. Probably the most popular application of PCA for feature extraction are the "eigenfaces" [Turk et al., 1991]. PCA, also known as Karhunen Loeve transform [Nixon et al., 2012b], can be used for dimensionality reduction. Applying it to reduce the dimensions of the output of another feature descriptor (or concatenations of multiple feature descriptors), also results in a feature description. For the "eigenfaces" PCA is applied to reduce the dimension of concatenated pixel values of a region (Section 2.3.1). Although PCA is a very powerful technique and it has been successfully applied for feature extraction, it has to be applied with care for unbalanced data, as components capture the highest variance. The highest variance of the data does not necessarily correlate with the discriminability of different classes in the feature space.

Data-driven feature computation is applied in Section 6.1.3 in form of CNNs.

2.4 Feature normalization

Feature normalization can have a significant influence on classification performances. If using multiple features with different ranges, this becomes most clear. Each element $v_j^{(i)}$ of an extracted feature vector

$$\mathbf{v}^{(i)} = \begin{pmatrix} v_1^{(i)} \\ \ldots \\ v_D^{(i)} \end{pmatrix}$$
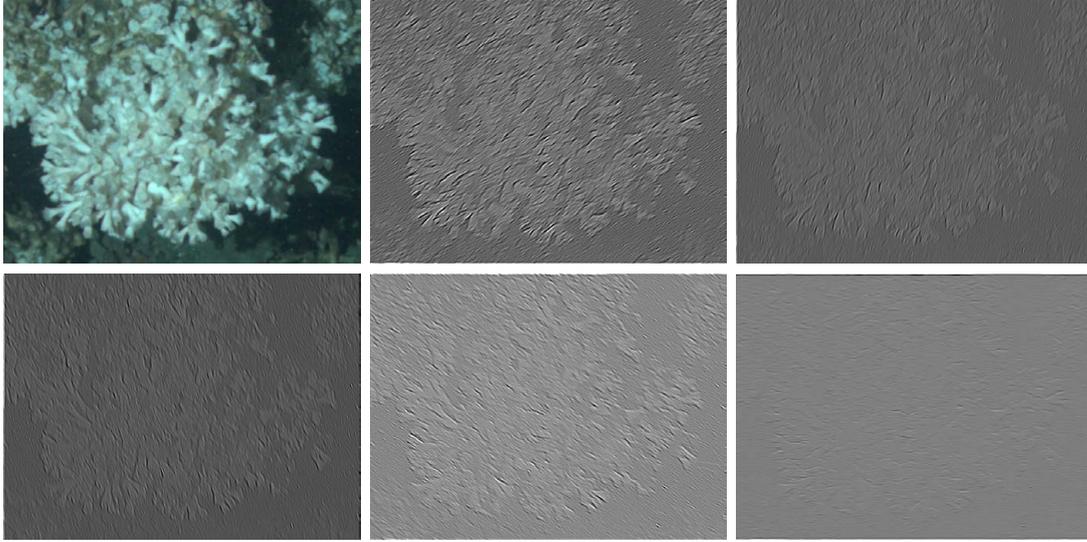
Figure 2.6: Original image (upper left) and Gabor filter responses for 5 orientations with scale = 3. The contrast and brightness is increased for the Gabor responses for visualization purposes.

with dimension D, needs to be re-scaled:

$$\hat{v_j}^{(i)} = \frac{v_n^{(i)} - \min\limits_{i' \in 1, \ldots, N} \{v_j^{(i')}\}}{\max\limits_{i' \in 1, \ldots, N} \{v_j^{(i')}\} - \min\limits_{i' \in 1, \ldots, N} \{v_j^{(i')}\}}$$

or even standardized [Bishop, 2006]:

$$\tilde{\mathbf{v}}^{(i)} = \frac{\mathbf{v}^{(i)} - \bar{\mathbf{v}}}{\sigma}$$

with $\bar{\mathbf{v}}$ being the average feature vector and σ the standard deviation of all $\mathbf{v}^{(i)}$.

Feature normalization prevents that elements v_j with higher variance will become more important for a classification system than others. Feature normalization can be applied for each feature descriptor or for each element position of the extracted feature vectors $\mathbf{v}^{(i)}$. Furthermore this normalization can be applied per image, all feature vectors from one image are then normalized individually. The need for feature normalization of course is dependent on the applied ML algorithm. Random Forests for instance do not need this scaling and CNNs often apply some kind of feature normalization by subtracting the mean image. Feature normalization is also applied by several image preprocessing algorithms (Chapter 5) in order to increase the color constancy throughout a time-series of recorded images.

2.5   Feature selection

Using high dimensional features spaces can be problematic due to the curse of dimensionality [Bellman, 1961; Hastie et al., 2009, Ch. 2.5]. With increasing dimensions, the volume of the feature space increases exponentially and the data coverage becomes sparse as distances between data-points highly increase. A reduction of the feature space dimension, can therefore not only reduce the computational costs for the applied ML algorithm, but can also increase the generalization of a classifier. The performance increases on unseen data, which was not used during training (more details in Section 2.8). The reduction of the dimensions can for instance be achieved by feature transformation methods such as PCA (Section 2.3.5) or by feature selection methods such as forward or backward selection. For the latter two, the performance of a classifier is evaluated for different sets of features. For the forward selection method features are added iteratively for instance until a certain classification performance is reached. For backward selection features are excluded iteratively as long as a certain classification performance will not drop below a chosen value. Some classifiers also provide feature importance evaluation like the Random Forests algorithm (Section 2.7.2).

2.6   Augmentation

Data augmentation, also called data boosting, is one method allowing to achieve sufficient classification results with a restricted amount of training examples. It allowed convolutional neural networks (see below) to beat humans in classification performance for the first time [He et al., 2016; Witten et al., 2016]. Furthermore, augmentation can be used to reduce the disadvantages of training with highly unbalanced training-sets, by applying higher augmentation factors for classes with less instances generating a balanced training-set. For CV, augmentation can for instance be achieved by rotating and flipping the training images and adding noise before feature extraction (Figure 2.7). The augmentation strategy needs to be adapted to the applied type of feature extraction as for instance histogram features (Section 2.3.2) or other local statistics (Section 2.3.1) are rotation invariant.

Data augmentation is one method to solve the AC challenge and is applied as described in Section 6.1.3. In cases where the feature extraction is limited to annotated regions in images, also a variation of the location of extraction can be applied to increase the amount of feature vectors. Feature vectors are extracted for instance on a 3×3 neighborhood with the same annotation class. This strategy is dependent on the chosen annotation strategy, e.g. chosen shapes (Chapter 3), and meta information in form of the spatial extent about the OOIs.

2.7   Machine learning

In this thesis, Machine Learning (ML) is referring to empirical learning automated by the use of computers, as defined in [Shavlik et al., 1990]. The idea is to generate rules from externally supplied examples, also called training. These examples are feature vectors $\{\mathbf{v}^{(1)}, ..., \mathbf{v}^{(N)}\}$,
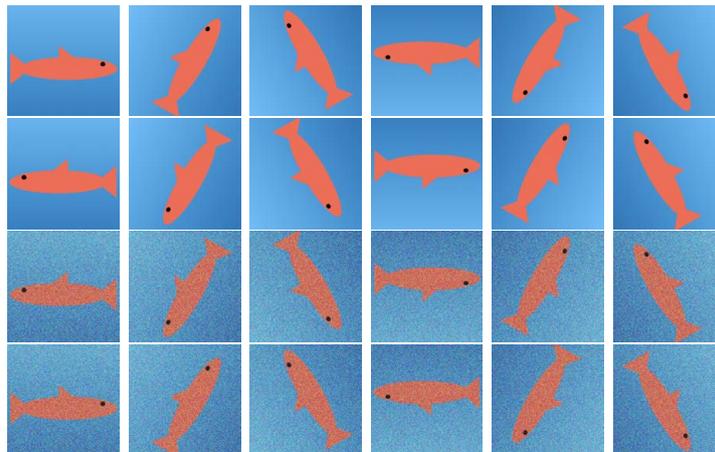
Figure 2.7: Data augmentation for an artificial example image. The image of a fish (upper left) is rotated in $60°$ steps (first row), flipped and again rotated (second row). Gaussian noise was added on both transformations (third and fourth row) to further augment the data.

here, extracted from images. Applying the learned rules is called classifying, which is the assignment of a class to a feature vector $\mathbf{v}^{(i)}$:

$$\Delta(\mathbf{v}^{(i)}) = y^{(i)}.$$

The outcome of the ML is therefore a trained classifier Δ.

In general, it can be distinguished between (i) unsupervised and (ii) supervised ML. For (i), no annotations are provided. Here, the idea is to find regularities and similarities to structure the learned data. Most unsupervised ML methods look for similarities by clustering the data [Shavlik et al., 1990, Ch. 1]. For (ii) the feature vectors used for the training of the classifier need to be annotated. For each $\mathbf{v}^{(i)}$, a class $\omega^{(i)} \in \Omega$ needs to be assigned. The corresponding ML algorithm tries to learn the best possible way to perform a mapping, which allows to distinguish between different annotation classes Ω in the most general way. Good rules should "capture the general patterns" [Shavlik et al., 1990].

For both, supervised and unsupervised ML, the provided data, i.e. the extracted feature vectors $\mathbf{v}^{(i)}$, are usually divided into training, evaluation and testing sets. The training-set is only used during the training step, learning the structure or to discriminate the annotation classes. The evaluation set is used to tune parameters of the ML algorithm and the test-set to finally test the performance on unseen data in order to predict the generalization of the learned model (see Section 2.8 for further details).

2.7.1 Unsupervised machine learning

The unsupervised ML algorithms applied in this thesis are all clustering algorithms. In contrast to supervised ML, no classes are pre-defined here. Usually only the number of classes K to

distinguish needs to be known or guessed, before applying the clustering algorithm. K is then referring to the number of clusters. At some point, a manual annotation step is also needed for applied unsupervised methods, but in contrast to supervised ML it is carried out after training the classifier.

In oder to understand the more complex clustering algorithms, first some basic algorithms are introduced, followed by the more advanced methods applied in this thesis.

K-Means K-means clustering [MacQueen, 1967; Bishop, 2006] is probably the best known clustering algorithm. As for most clustering algorithms, the important parameter is the number of clusters K.
First, initial prototypes, i.e. the cluster centers, $\{\mathbf{u}^{(k)}\}_{k=1,\ldots,K}$ are chosen by randomly picking K feature vectors from the training data.
Second, for each feature vector $\mathbf{v}^{(i)}$, the closest $\mathbf{u}^{(k)}$ is determined:

$$\mu(\mathbf{v}^{(i)}) = \mathrm{argmin}_k \left\{ d(\mathbf{v}^{(i)} - \mathbf{u}^{(k)}) \right\}, \tag{2.1}$$

with $d(\ )$ being a distance measure like the Euclidean metric ($||\mathbf{v}^{(i)} - \mathbf{u}^{(k)}||$). The closest prototype $\mathbf{u}^{(k)}$ for a feature vector $\mathbf{v}^{(i)}$ is also its best matching unit (BMU).
Third, after all $\mathbf{v}^{(i)}$ are assigned to one $\mathbf{u}^{(k)}$ all $\mathbf{u}^{(k)}$ are updated by:

$$\mathbf{u}^{(k)} = \frac{\sum_i \delta_{\mu(\mathbf{v}^{(i)}),k} \cdot \mathbf{v}^{(i)}}{\sum_i \delta_{\mu(\mathbf{v}^{(i)}),k}}$$

with δ being the Kronecker delta. $\mathbf{v}^{(i)}$ $\mathbf{u}^{(k)}$ assigning and updating the $\mathbf{u}^{(k)}$ is repeated until no further change in the $\mathbf{v}^{(i)}$ $\mathbf{u}^{(k)}$ assigning is made or a maximum number of iterations is exceeded.

Self Organizing Map   In order to understand the principle of the hierarchical hyperbolic self organizing map (H^{2}SOM) algorithm, first the self Organizing Map (SOM) algorithm [Kohonen, 1989] is explained. In contrast to the K-means algorithm, the prototypes of the SOMs are organized in a topology independent from the feature space. The topology has a net like structure (see Figure 2.8). In the update step, a shifting of a prototype also affects its neighbor prototypes in the SOM topology. Using connected prototypes was inspired by the connection of neurons in the human brain. SOMs can have different topologies describing the connections of the prototypes. The simplest topology is a regular two-dimensional (2D) grid. In contrast to the K-means algorithm, the prototypes are updated after each assignment of a feature vector $\mathbf{v}^{(i)}$ to a prototype $\mathbf{u}^{(k)}$ with $k = \mathrm{argmin}_j \left\{ d(\mathbf{v}^{(i)} - \mathbf{u}^{(j)}) \right\}$. $\mathbf{v}^{(i)}$ are chosen randomly and updating the corresponding BMU $\mathbf{u}^{(k)}$ also results in adapting connected neighboring prototypes $\mathbf{u}^{(k')}$. The influence of adaption is usually modeled by a Gaussian distribution, so that a $\mathbf{u}^{(k')}$ is adapted stronger, the closer it is to $\mathbf{u}^{(k)}$ in the SOM topology. Furthermore, this adaption rate is decreasing throughout the training process to prevent high adaption of prototypes at the end of the training.
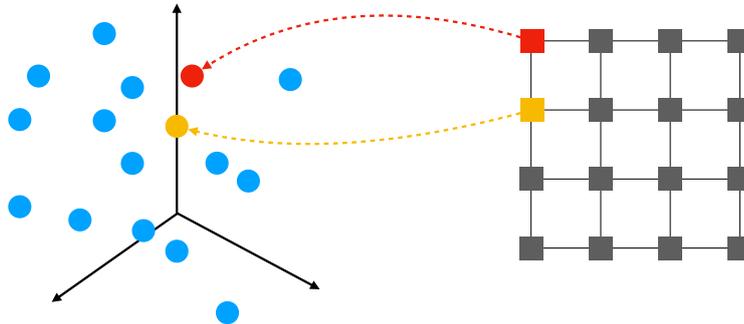
Figure 2.8: The topology, a regular 2D grid, of a Self Organizing Map (SOM). The topology of the SOM is independent from the feature space. For two prototypes their position in the feature space and position in the SOM topology is highlighted (red and yellow).

Hierarchical Hyperbolic Self Organizing Map   The hierarchical hyperbolic self organizing map (H²SOM) [Ontrup et al., 2006] is also a SOM but has a hyperbolic topology and is trained hierarchically in rings. The topology is defined by the number of rings and neighbors. Typical values are three rings and eight neighbors and such an H²SOM contains $K = 161$ prototypes. The initial ring, not counted here, consists of only of one prototype, the center of the training data in the feature space. The number of prototypes of the following rings is depending on the ring position and number of neighbors defined for the training. For three rings and eight neighbors the first ring consists of eight prototypes. Each ring is trained individually and the adaption of prototypes is therefore only carried out on the currently trained ring. For new rings prototypes are placed at the position of its parent prototype from the previously trained ring. The number of prototypes is usually higher for following rings as a parent prototype has several child prototypes. The topology and hierarchical training allows to use a fast beam search to find the corresponding BMU of a feature vector $\mathbf{v}^{(i)}$ at lower computation cost. Starting at the initial ring, the BMU search is reduced to connected prototypes of the next ring, the child prototypes of the BMU of the current ring. This search strategy is applied until the BMU on the last outer ring is identified.

Using a hyperbolic topology, allows to visualize the prototypes on HSV color space disc. The hyperbolic geometry can be mapped to a 2D circle, giving each $\mathbf{u}^{(k)}$ a unique color (Fig 2.9). This visualization can be used to manually explore the learned clustering of feature vectors extracted from an image (see Figure 2.9). Prototypes close in the topology are mapped to similar colors. This colorization can of course be changed dynamically for instance to focus on certain prototypes increasing the color contrast between them.

The H²SOM algorithm is applied in Section 6.1.2 for the segmentation of a cold water coral colony in images recorded *in-situ*. Furthermore, it is applied to segment live calcareous algae on images recorded for a *wet-lab* study. For both applications, the corresponding H²SOMs are trained with 3 rings and 8 neighbors, resulting in H²SOMs with 161 prototypes. The
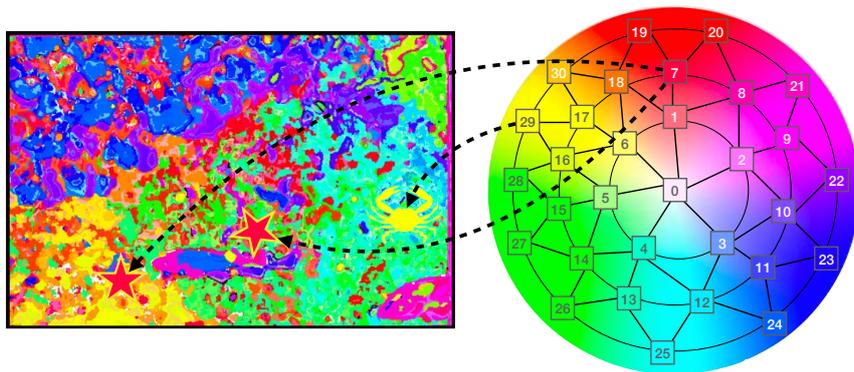
Figure 2.9: The Hierarchical Hyperbolic Self Organizing Map (H²SOM). On the right the 2D projection on an HSV disc of the hyperbolic topology is shown and on the left an H²SOM colorization of a clustered image is shown. Image pixels are colored by the BMU of the corresponding feature vector $\mathbf{v}^{(i)}$ extracted on $\mathbf{I}(x, y)$. For illustration purpose two artificial starfish and one crab are added to the image on the left.

prototype adaption is decreased in the training process using a an exponentially decreasing learning rate. Both for training and classification an Euclidean metric was applied.

Super-pixel   The idea behind super pixels is to group similar neighboring pixels. This pre-segmentation allows to speed up a fully segmentation and / or classification. Several super-pixel algorithms exist like [Felzenszwalb et al., 2004; Shi et al., 2000; Levinshtein et al., 2009; Veksler et al., 2010]. In this thesis, the Simple Linear Iterative Clustering (SLIC) super-pixel algorithm [Achanta et al., 2010; Achanta et al., 2012] is applied, which is an adaption of K-means clustering algorithm. Pixel color similarities are computed in the CIE L*a*b* color space, as it allows to use the Euclidean distance measurement to estimate color differences. K prototypes are initialized using a regular grid with a step-size s. For this step the locations of the cluster prototypes are improved by finding the lowest gradient position in a 3×3 neighborhood (η_3) to avoid initial positioning on very noisy pixels or edges. As in the K-means algorithm, for each pixel $\mathbf{I}(x, y)$ the closest prototype (BMU) is computed considering color similarities d_c and spatial distance in the image d_s by using a combined metric with a weight factor w_{sp} for controlling the importance of d_c and d_s:

$$d^* = \sqrt{d_c{}^2 + \left(\frac{d_s}{s}\right) w_{\mathsf{sp}}}.$$

As pixels should only be grouped in a local region the searching for a BMU is only carried out in a $2s \times 2s$ region around $\mathbf{I}(x, y)$. Next using the estimated BMUs each centroid is updated for position and value. The last two steps are repeated until a residual error (L_2 norm) is below a certain threshold or if the number of iterations exceeds an iteration threshold. The algorithm converges so fast that typically 10 iterations are enough.
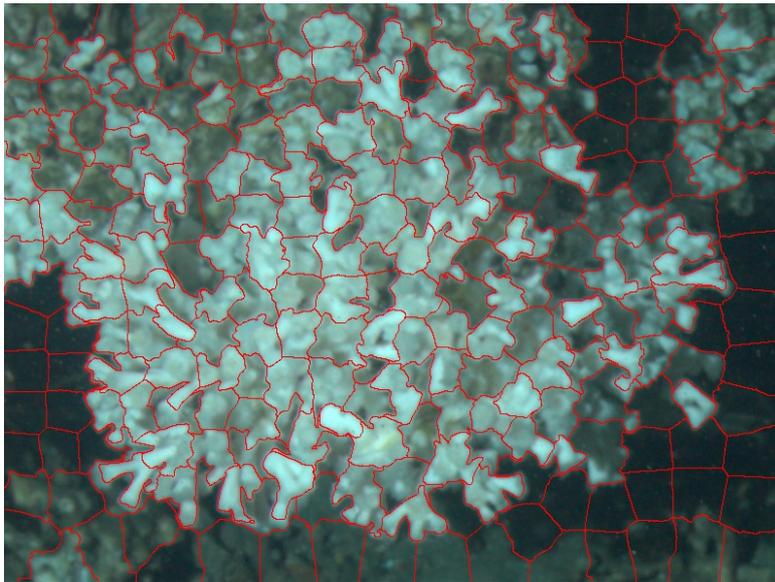
Figure 2.10: The Simple Linear Iterative Clustering (SLIC) super-pixel algorithm. Super-pixels are outlined in red. It can be noticed that the structure of the coral branches is segmented by different super-pixels, allowing to reduce the segmentation of the coral to a classification of single super-pixels.

The super-pixel algorithm is applied in Section 6.2 to pre-segment the processed images. The algorithm is applied with $w_{\mathsf{sp}} = 40$ and a step size of $s = 20$ pixels. A visualization of the result of applying SLIC is shown in Figure 2.10.

2.7.2 Supervised machine learning

The supervised ML algorithms applied in this thesis are described in this Section. In contrast to the unsupervised ML algorithms, annotated data in form of pairs of feature vector and annotation class $(\mathbf{v}^{(i)}, \omega^{(i)})$ are needed here. The task of assigning a class $\omega^{(i)} \in \Omega$ is usually performed by a manual annotation process. For underwater images, this manual annotation process can be a challenging task and is described in detail in Chapter 3.

In oder to understand the more complex supervised ML algorithms, first some basic algorithms are introduced, followed by the more advanced methods applied in this thesis.

Classification and Regression Tries The Classification and Regression Trees (CART) algorithm [Breiman et al., 1984; Bishop, 2006] can be used for classification and for regression. A CART is a binary decision tree, where at each node one specific variable $v_j^{(i)}$ of the input feature vector $\mathbf{v}^{(i)}$ of dimension D with $\{j\} \in \{1, ..., D\}$, is compared with a threshold l. Whether the value of the variable is below it or not the path in the tree is chosen. Thus, a split is separating feature vectors at a node as the continuing path is defined by the split

criteria. This is repeated for every node on the path until an endpoint, i.e. a leaf, is reached and each leaf defines a classification output.

For training of a CART, best split criteria need to be identified, which is done hierarchically. Starting with the root node, only one node is added at a time. Among all variables $\{j\} \in \{1, ..., D\}$ the best split variable and the related threshold l is identified by computing a misclassification score. The misclassification score is computed for both following nodes on the corresponding training data following the specific path in the tree. Only feature vectors $\mathbf{v}^{(i')} \in \Theta$ following this specific path are considered. For finding the best split three different misclassification scores can be considered. Several misclassification scores exists.

The first is the misclassification error. It uses the majority class of Θ (ω_Θ) to simulate the classification output for a chosen split:

$$\frac{1}{|\Theta|} \sum_{\mathbf{v}^{(i)} \in \Theta} \omega_\Theta \neq \omega^{(i)}$$

The other measurements are the Gini index, computed as:

$$\sum_{\omega \neq \omega'} p_{\Theta,\omega} p_{\Theta,\omega'} = \sum_{\omega=1}^{\Omega} p_{\Theta,\omega}(1 - p_{\Theta,\omega})$$

and the cross-entropy, computed as:

$$-\sum_{\omega=1}^{\Omega} p_{\Theta,\omega} \log p_{\Theta,\omega}$$

with

$$p_{\Theta,\omega} = \frac{1}{|\Theta|} \sum_{\mathbf{v}^{(i)} \in \Theta} \omega^{(i)} = \omega$$

for all classes Ω. The Gini index and the cross-entropy are often more suitable, as they generate regions in the feature space which are more pure of one class, meaning regions which have a high amount of data from one class. Another advantage is: they are differentiable and therefore suitable for gradient based optimizations.

The most obvious advantage of the CARTs is the simplicity of the interpretation of this classification system. Threshold choices on variables can be modeled by a simple decision tree diagram. Also it is simple to extract classification rules from these CARTs which can be handled very well by humans. The advantage of having only very few training parameters is to be mentioned as well. But the simplicity also leads to disadvantages. Trees tend to be very sensitive to the data that are learned, so a minor change of the training data might generate a very different tree with other splits and thresholds. Therefore, trees show a tendency to over-fit the training data. It is a low bias and high variance classifier. Another problem is, splits are always aligned to the axes of the feature space, so classes which can be separated most perfectly by a 45 degrees line to the axes need very many nodes to be separated [Bishop, 2006; Hastie et al., 2009].

Random Forests The Random Forest (RF) algorithm [Breiman, 2001; Hastie et al., 2009, p. 588] is an improvement of CARTs. It is an ensemble learning method using many weak learners in form of CARTs to generate a system more intelligent than each individual learner. In a "forest" all trees are trained differently and the classification output is computed as the majority vote of all trees in the forest (Figure 2.11). This method aims to reduce the high variance, i.e. the over-fitting problem of the trees. For the training of each tree a new randomly chosen subset of features as well as a new randomly chosen subset of data is selected, allowing re-selection. The size of the subset of features is usually the square root of the dimension D of $\mathbf{v}^{(i)}$ and is equal for all trees. The subset of the data which is not used for training, usually one third of the data, is used to calculate individual classification errors, the so called out-of-bag (OOB) error. As the training data for each tree b varies, also the OOB data varies for each tree. For training a RF, new trees will be added until a predefined maximal number of trees in the forest is reached or if the OOB-error drops below a value which is to be set for the training. The features for training a tree directly influence the de-correlation between the trees. If the number of features is very high, the features selected as a split criteria can be very similar. If the maximal features value is very low the features used in the tree differ much hence the trees differ much. All these training parameters need to be tuned to avoid over-fitting of the training data.

In data-mining often not all variables are equally relevant. A good classification can be done with only a small subset of the variables. The RF algorithm comes with a calculation of the variable importance [Hastie et al., 2009], which can also be used for feature selection (Section 2.5). This calculation makes use of the out-of-bag (OOB) samples. After the training of a tree b in the forest the corresponding out-of-bag error OOB_b is computed. The values for the jth variable in the OOB samples are replaced with randomly permuted values and a new out-of-bag error $OOB_{b,j}$ is computed. The variable importance $v(j)$ of the jth variable is then computed as:

$$v(j) = \frac{\hat{v}(j)}{\sum_j \hat{v}(j)}$$

with

$$\hat{v}(j) = \frac{\sum_{b=1}^{N} \text{OOB}_{b,j} - \text{OOB}_b}{N}$$

. The $v(j)$ measures therefore, how the classification result would be affected, if the variable would be random.

The RF classifier benefits of the simplicity of the CART model. It can handle multi class classification directly. Furthermore, it can handle categorical values as well as numerical values and its classification performance is rather comprehensible. RFs and CARTs have the same ability to handle missing values using surrogate splits, using alternative split features. The RF makes use of the OOB-error to reduce the problem of high variance of the CARTs but not all disadvantages of the CARTs are eliminated. The splitting in the nodes is still always aligned to the axes of the feature space and hence features that can be splitted at a 45 degree axis require very deep trees.
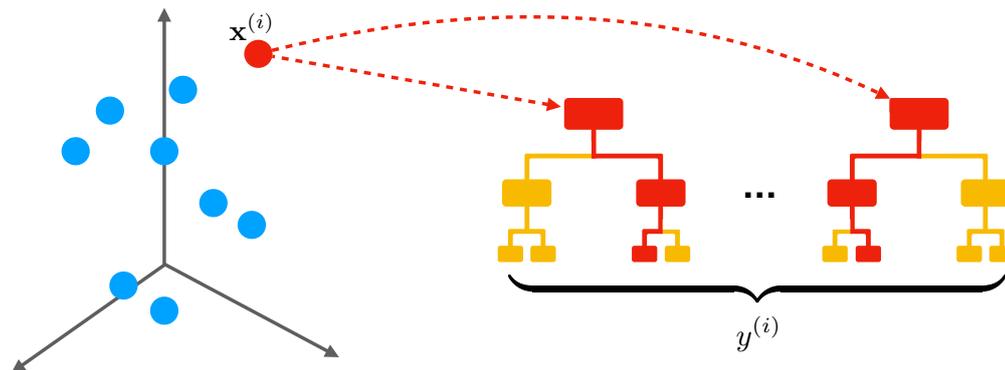
Figure 2.11: The Random Forest (RF) algorithm. The RF consists of several CARTs all trained with different training-sets, resulting in different CARTs. Each data point, i.e. a feature vector $\mathbf{v}^{(i)}$, is classified by each tree. The classification output of the RF $\Delta(\mathbf{v}^{(i)}) = y^{(i)}$ is computed as the majority vote of all CARTs in the forest.

The RF algorithm is applied in Section 6.2 to detect shrimp instances in images recorded *in-situ*. Training parameters of the RF were tuned using four-fold cross validation (Section 2.8).

2.7.3 Deep learning

Deep learning algorithms can either be supervised or unsupervised [Deng et al., 2014]. Deep learning has become very popular recently, due to the increased computational power that comes with to use of general purpose graphical processing units (GPGPUs), significantly increased amount of available training data and recent scientific advances of the algorithms [Deng et al., 2014]. Deep learning includes a wide spectrum of algorithms able to solve different ML problems. The focus of this section will be on supervised deep learning, more precisely on convolutional layer networks, which have been applied successfully to CV tasks.

Multi layer perceptrons In order to understand Convolutional Neural Networks (CNN) first the Multiple Layer Perceptron (MLP) is explained briefly which forms the basis of CNNs [Deng et al., 2014; Goodfellow et al., 2016]. MLPs consist of nodes, i.e. the neurons, organized in a multi layer structure. Neurons of one layer are only connected to the neurons of the following layer. Thus, MLPs are also called feed forward networks, as the information flow is only forward throughout all computations from input to output. The number of layer defines the depth of the network. The first layer is called input layer and the final layer the output layer. As the MLP is a supervised ML algorithm, a function that maps the input $\mathbf{v}^{(i)}$ to the desired output $\omega^{(i)}$ needs to be found:

$$\Delta(\mathbf{v}^{(i)}) = y^{(i)} \approx \omega^{(i)}$$

input layer   hidden layer   output layer
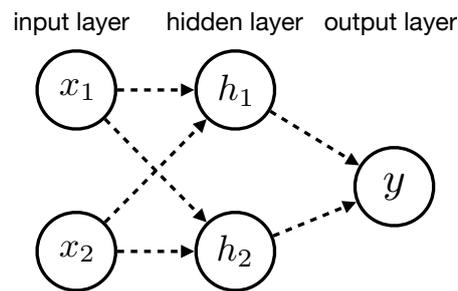
Figure 2.12: Multi layer perceptron with input and output layer and one hidden layer.

The training data only defines for each input $\mathbf{v}^{(i)}$ the desired output $\omega^{(i)}$, but not the hidden layer (Figure 2.12). The hidden layer can be compared to the kernel trick for SVMs [Cortes et al., 1995], the input is transformed by nonlinear function(s). In MLPs all neurons of one layer are connected to all neurons of the following layer, thus they are fully connected networks. The minimal number of layers is three, including input and output layer. An MLP has therefore at least one hidden layer but can have several further hidden layers creating a deep network. The hidden layers can have different activations functions. Typical activations are rectified linear units $g(z) = \max(0, z)$ applied on top of an affine transformation $\mathbf{h}(\mathbf{v}) = g(W^T \mathbf{v} + \mathbf{b})$. Other typical activation functions are logistic sigmoid or the hyperbolic tangent activations.

For training the network, weights of the connections and parameters of the transformation functions need to be tuned. This tuning can be performed using gradient based learning on a loss function [Lecun et al., 1998]. The loss function, or cost function can be the mean squared error between $\{y^{(i)}\}$ and $\{w^{(i)}\}$. A well known gradient based learning method is back-propagation [Rumelhart et al., 1988]. The basic idea of back-propagation is to compute the gradients by propagation from the output layer to the input layer [Lecun et al., 1998].

Convolutional Neural Networks   Convolutional neural networks [Lecun et al., 1998; Deng et al., 2014; Goodfellow et al., 2016, Ch. 9] are feed forward networks like MLPs, but have at least one convolutional layer. This convolution is similar to "normal" convolutions with a kernel function, e.g. convolution with a Gauss kernel for image blurring. The kernel is typically smaller than the input creating sparse interactions e.g. to detect edges in images. Convolutions are realized using weight-sharing, so the perceptual features like edges can be detected at different positions in the input image. Neurons of a layer sharing the same weights are grouped as feature maps. The convolutional layer typically consists of several convolutions so that multiple features can be detected at all image locations. Usually a nonlinear activation function like the rectified linear unit (see above) follows the convolution, to create activation or feature maps. The feature maps can be interpreted as the responses of different feature extractions. Convolution and applied activation functions are usually grouped in one convolutional layer. The following layer is usually a pooling layer, applying pooling functions like max pooling, which outputs the maximum within a rectangular neighborhood
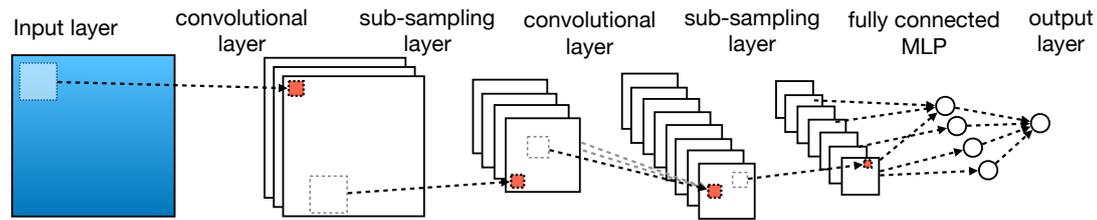
Figure 2.13: A convolutional neural network with a LeNet-5 like layout. The displayed network consists of two convolutional layers, two sub-sampling layer, a fully connected layer (MLP) and an output layer.

to reduce the size of the convolutional layer output. This combination of convolutional layer and pooling layer can be connected in series to create deep networks. The final classification output is computed by a fully connected layer similar to MLPs. A visualization of a CNN with described layer is presented in Figure 2.13.

CNNs have been applied very successfully and have even outperformed humans in an image detection task [He et al., 2016]. One reason for the usually very high accuracy of CNN is the ability to learn feature representations which allow to discriminate the different annotation classes. The weights of neurons are learned creating different convolutions. The convolutions themselves are therefore adapted to the training data and thus the feature extractions are data-driven. Probably the biggest disadvantage is the need for a high amount of training data as CNNs have a huge number of training parameters. Thus, CNNs show a tendency to over-fitting the training data, if the amount of the latter is too low compared to the number of parameters to be tuned.

As CNNs can consist of several layers which are connected following a specific layout. Several layouts exist which can be categorized by their number of convolutions and depth. In the following the LeNet-5 layout [Lecun et al., 1998] is described in detail as it is applied in Section 6.1.3.

LeNet-5 The CNN used in this thesis uses the layout of the LeNet-5 CNN [Lecun et al., 1998]. The layout of the network with its corresponding layers is visualized in Figure 2.14. The network consists of only 7 layers not counting the input layer. First a normalization is applied to all pixels values of the input image. The first layer, a convolutional layer (conv1) is applied with a kernel size of 5 and 20 feature maps corresponding to 20 feature representations. The connected pooling layer (pool1) applies a max pooling with a neighborhood of 2×2 reducing the feature map's sizes of the convolutional layer by 2. The next layer is again a convolutional layer (conv2) and has again a kernel size of 5 but has 50 feature maps. The output of this convolutional layer is again reduced in a pooling layer (pool2) by factor 2. The last convolutional layer (conv3) might also be referred to as a fully connected layer, as the kernel size is equal to the input size (both 5×5) resulting in 500 feature maps with size 1×1. The last layer before the output layer is again a fully connected layer (fullyc) containing
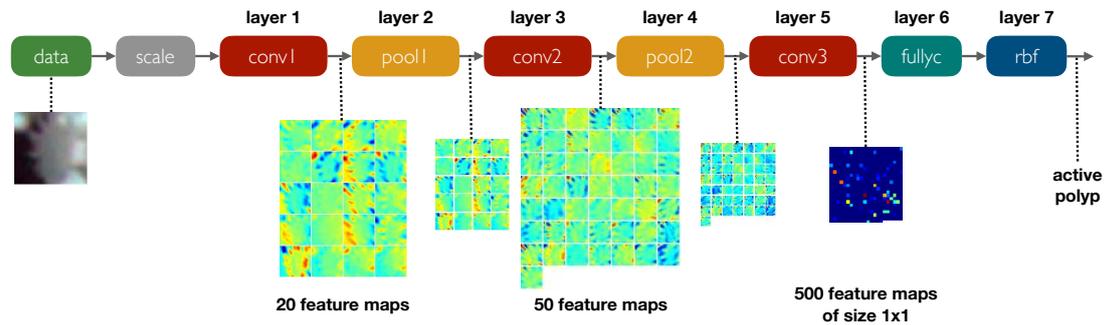
Figure 2.14: The convolutional layer network with a LeNet-5 layout trained to classify cold water coral polyp states (Section 6.1.3). The network consist of 7 layer not counting the input layer. Convolutional layers are color coded in red and pooling layers are color coded in yellow. The outputs of the convolutional and pooling layers are visualized for the input image on the left showing a coral branch with active polyps. Activations range from red for high responses to blue for low responses. Looking at the convolutional layer (conv1), it can be noticed that one feature map (lower left) seems to highlight the polyps in the image. The input image is classified as "active polyp".

84 neurons. The final classification output is computed similar to the MLPs (see above) to create the classification output from a hidden layer. Here, the final layer is computed from Euclidean radial basis functions of the form:

$$y_i = \sum_j (z_j - w_{ij})^2$$

with w_{ij} as the weights of this final layer and z_j as the output from the previous layer (fullyc). The network is trained using back-propagation with a mean squared error as the loss function.

All in all for this network about $20,000$ parameters need to be trained making it a rather small network compared for instance to the GoogLeNet [Szegedy et al., 2015] with more than 6 million parameters and 22 layers.

In Section 6.1.3 a CNN with the described LeNet-5 layout is trained and applied to classify the activity state of cold water coral polyps.

2.8 Training and tuning

One problem for training a classifier is over-fitting of the training data. The classifier does not generalize well, as it perfectly learns the training data but does not learn functions which allow to classify slightly different data. A common way of overcoming this problem is to divide the available dataset into a training, a validation- and / or test-set. The parameters of a ML algorithm are tuned using the training- and validation-set by learning the training-set and validating the generalization on the validation-set by computing the classification performance (see Section 2.10) on the validation-set. For small amounts of data an S-fold

Figure 2.15: A four-fold ($S = 4$) cross-validation-set. Each data element is included only in one training and validation run (numbered from 1 to 4). In each run the validation-set (colored in green) is used for optimizing the training parameters, while the rest of the data (colored in blue) is used for training.

cross validation [Bishop, 2006] can be applied to reduce the dependence on a well chosen validation-set. For a dataset of N elements, $(S-1)/S \cdot N$ elements are used for training and the rest for validation. This splitting is performed S times such that each data element is in the validation-set exactly in one run (see Figure 2.15). The parameters of the applied ML algorithm can then be tuned to perform best in each run on the corresponding validation-set. The best parameter set can then be used to train a classifier with training and validation data. The final test is performed on the test data, to predict the performance on unseen data as no optimization of the ML algorithm was performed for this data. The performance on the test data shows the generalization of the learned class-differentiation.

2.9 Post-processing

In general all processing performed after the classification by the trained ML algorithm is considered as a post-processing. The output of a classifier often is not a pure class-label but a probability value for each class. The output can therefore be tuned to be more generous or strict for a specific class. Some ML algorithms already consider this during the learning process like the RF algorithm. It uses class weights to adapt the cost for a misclassification of a specific class over another.

In CV post-processing often needs to be applied to improve the performance on *full images*, as the included ML algorithm is only trained on image regions (Section 2.3). The type of post-processing is dependent on the corresponding task of the CV system. This can range from identifying the occurrence of a certain class of OOIs in an image ignoring the number of occurrences per images, to the full segmentation of a class of OOIs in all images to investigate the growth or shrinking of a species of interest.

As stated earlier, feature vectors $\mathbf{v}^{(i)}$ are generated from image regions and can therefore be localized in the image of extraction. This also accounts for the classification $\Delta(\mathbf{v}^{(i)})$ and can
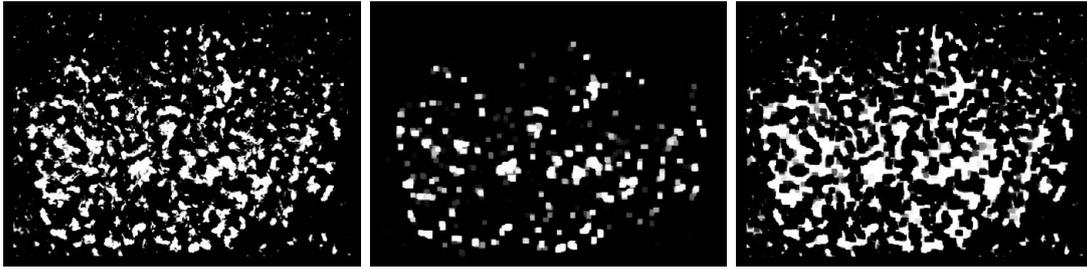
Figure 2.16: The post-processing of a classification output. The classification output is interpreted as a gray scale image $\mathbf{I}^{(\Delta)}$ (left). The classification output is encoded as the class probability of one class. The probability is encoded in the pixel intensity, so high class probabilities are white and low class probabilities are dark. In the middle and on the right, two post-processed versions of $\mathbf{I}^{(\Delta)}$ are shown. In the middle the morphological opening operation $\phi^{(\text{open})}$ was applied and in the right, the morphological closing $\phi^{(\text{closing})}$ operation was applied.

be used to generate a classification output image $\mathbf{I}_t^{(\Delta)}$ for each $\mathbf{I}_t$. The pixel values $\mathbf{I}^{(\Delta)}(x, y)$ of the output image correspond to the classification output of the feature vector $\Delta(\mathbf{v}^{(i)})$ located at (x, y). In order to tune the CV task morphologic operations, like combinations of dilation and erosion on such an output image can be applied as shown in Figure 2.16.

Different post processing are applied at each of the presented CV applications (Chapter 6-7).

2.10 Evaluation

The evaluation of the performance of a CV system often needs to be adapted to the purpose of the application similar to the post-processing. The questions one might ask for instance are:

- What is the field of application?

- Are the costs for a misclassification for all classes equal?

- Is a precise segmentation on a pixel of an OOI needed and possible?

Furthermore, it is dependent on the available ground truth.

As a CV system can consist of several processes, different stages can be evaluated and tuned. Usually for each evaluation step true positives (TP), true negatives (TN), false positives (TP) and false negatives (TP) need to be defined. For CV systems using supervised ML on a two class problem ($\omega \in \{0, 1\}$) the evaluation can be done straightforward. The annotation classes $\{\omega^{(i)}\}$ of corresponding feature vectors $\{\mathbf{v}^{(i)}\}$ are compared with the classification outputs $\{\Delta(\mathbf{v}^{(i)})\}$. For two class problems, classes are typically referred to as a positive class $\omega = 1$ and a negative class $\omega = 0$. A TP is then counted if $\Delta(\mathbf{v}^{(i)}) = \omega^{(i)} = 1$, a FP if $\Delta(\mathbf{v}^{(i)}) \neq \omega^{(i)} = 0$, a TN if $\Delta(\mathbf{v}^{(i)}) = \omega^{(i)} = 0$ and a FN if $\Delta(\mathbf{v}^{(i)}) \neq \omega^{(i)} = 1$. This can

of course also be extended to multi-class problems (see below). Having TP, FP, TN and FN defined usually the accuracy (ACC) is computed to evaluate the system and to compare it to other trained classifier.

$$\text{ACC} = \frac{|TP| + |TN|}{|TP| + |FP| + |TN| + |FN|} \tag{2.2}$$

But ACC is not always a good indicator for the performance of a classifier. For an evaluation with highly unbalanced data, the number of instances of one class by magnitudes higher than another class, recall and precision are more suitable. A typical example is a two class problem with one class being the background of an image and the other a scarcely occurring OOI.

$$\text{precision} = \frac{|TP|}{|TP| + |FP|} \tag{2.3}$$

$$\text{recall} = \frac{|TP|}{|TP| + |FN|} \tag{2.4}$$

Another score which can be useful for such unbalanced situations is the false positive rate (FPR):

$$\text{FPR} = \frac{|FP|}{|FP| + |TN|} \tag{2.5}$$

This score allows to evaluate the performance if a negative class, e.g. a background class, is only annotated rarely. The FPR score allows to investigate the performance on these explicit negatively annotated regions. Lower FPR scores stand for better classification performances. A similar score to FPR score is the true negative rate (TNR). It is defined as $\text{TNR} = \frac{|TN|}{|TN| + |FP|}$ and is complementary to the FPR: $\text{FPR} = 1 - \text{TNR}$.

One drawback of using for instance recall and precision and not the ACC is that two values need to be optimized influencing each other. A classifier which classifies everything as the positive class $\Delta(\mathbf{v}^{(i)}) = 1$ will result in recall $= 1$ and precision $= 0$ and a classifier which classifies nearly everything as the negative class $\Delta(\mathbf{v}^{(i)}) = 0$ will result in recall $= 0$ and precision ≈ 1, which is of course not an ideal classifier. The F-score can be the solution to this problem:

$$\mathsf{F}_\beta \text{ score} = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}} \tag{2.6}$$

Common β values are 1 for the harmonic mean between precision and recall, $\beta = 2$ to weight recall higher than the precision and $\beta = 0.5$ to weight the precision higher than the recall.

This can of course all be extended to multi-class problems, for instance by looking at confusion matrices and computing the evaluation scores class wise. TP, FP, TN, FN need to be computed class wise as $\text{TP}_\omega, \text{FP}_\omega, \text{TN}_\omega$ and FN_ω. The class wise ACC, for instance, is then computed as:

$$\text{ACC}_\omega = \frac{|\text{TP}_\omega| + |\text{TN}_\omega|}{|\text{TP}_\omega| + |\text{FP}_\omega| + |\text{TN}_\omega| + |\text{FN}_\omega|}.$$

The averaged class wise accuracy can then be used to describe the performance of the classification system:

$$\mathsf{ACC} = \frac{\sum_{\omega=1}^{|\Omega|} \mathsf{ACC}_\omega}{|\Omega|}.$$

For details see [Sokolova et al., 2009].

For CV systems at one point or another spatial image information needs to be taken into account. As described in Section 2.9 a classification result can also be interpreted as an image $\mathbf{I}^{(\Delta)}$. The evaluation can therefore also be carried out on those $\mathbf{I}^{(\Delta)}$. This also allows to consider the AU challenge in the evaluation. As stated earlier, this evaluation needs to consider the field of application of the CV system and is therefore explained in detail for each CV system separately in the corresponding Sections (Section 6.1, Section 6.2 and Chapter 7).

3 Manual annotation of underwater images

This section is based on publication 4. Notation and extent is adapted to the scope of this thesis.

Underwater images by themselves generate only very limited knowledge. The content of the images needs to be quantified so that it can be used in scientific analysis or in decision making processes. The image content is often identified initially by manually annotating these images, i.e. marking objects of interest (OOIs) in the images and a simultaneously assigning of class labels to the OOIs. The annotations are then either used directly to describe the image content or as the ground truth to apply machine learning, to create CV systems. Manual and machine learning based annotations can then be handled as nominal, qualitative and quantitative data in further analysis.

Placing annotations in underwater images manually, is a challenging process [Schoening et al., 2016b]. The underwater environment differs from the humans natural environment, increasing the complexity of the annotation task. Furthermore, the annotation has to be made using only the image itself, as very limited additional information supporting the annotation process is available. It is impossible to take another picture of the imaged species for instance from another angle. Additionally the image acquisition challenges (OP challenge, DOP challenge) affect the images.

In contrast to other scientific imaging domains, such as digital pathology, there are no protocols and guidelines available for the process of manually annotating underwater images. An initial approach to generate such a guideline is presented in Paper [Schoening et al., 2016b; Durden et al., 2016a] and will briefly be described here. It is essential to increase the reliability and to assess the quality of manual annotations especially in order to overcome or limit the effects of AQ challenge and AU challenge. Furthermore, annotations of high quality are essential to train reliable CV systems. An annotation is defined mathematically in Section 3.1, followed by the description of the evaluation methods for the manual annotations in Section 3.2. Software tools are needed to optimize the process of manually annotating underwater images. The software tools applied for manual annotation in this thesis are described briefly in Section 3.3.

3.1 Manual annotation

A manual annotation $a^{(i)}$ consists of a region $\mathbf{r}^{(i)}$, a class label $\omega^{(i)}$, and an observer $o^{(i)} \in O$:

$$a^{(i)} = (\mathbf{r}^{(i)}, \omega^{(i)}, o^{(i)}).$$

The latter is the person who created the annotation. A region $\mathbf{r}^{(i)}$ is defined as a subset of pixels from an image $\mathbf{I}_t$, often defined by a shape and a position.

It can be differentiated between multiple tasks of image annotation: full image annotation (FIA), whole image screening (WIS) and single patch confirmation (SPC). In FIA a class
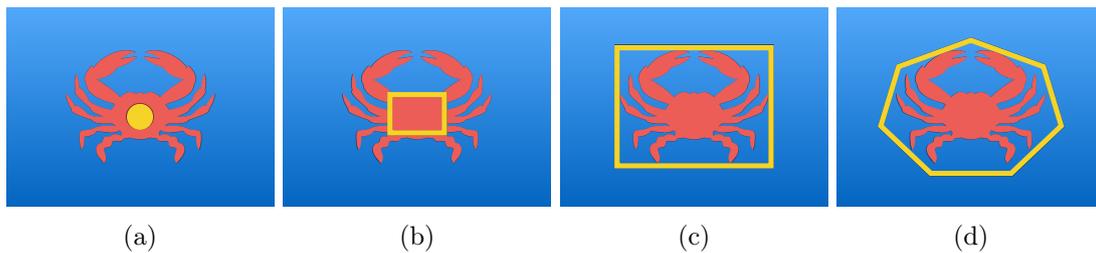
Figure 3.1: Different shape markers (yellow) for the annotation regions. Starting on the left (a), a simple point shaped annotation is used to annotate an artificial crab. Next, a rectangular regions are used to annotate the crab with two different strategies. First, the biggest rectangle only marking crab pixels (b) and second the smallest rectangle enclosing all crab pixels (c) is used. Finally, a polygon shape (d) is used to mark all crab pixels with a lower spatial uncertainty then (a),(b) and (c).

label is assigned to a full image. FIA is usually a very fast annotation type and is therefore often applied when annotation time is limited and a rapid annotation process is necessary as for instance in a live video stream. Furthermore it can be very useful in a initial screening process to filter out low quality images.

In WIS an observer has to search all relevant OOIs in an image and mark them with a region description $r^{(i)}$ and a class label $\omega^{(i)}$. This is the typical task for creating a set of manual annotations for ML in this thesis. WIS can be very time consuming and tiring. Defining regions is often reduced to simple shapes like circles, rectangles or even points to speed up the process. A most accurate manual annotation on a pixel level is most often impossible and the use of a computer mouse also comes with limitations. The spatial uncertainty introduced by simplifying the outlining of a region can limit the research output statement and has therefore to be adjusted to the research question in the first place. In an abundance study (Section 6.2), point annotations can provide precise enough information, while in a study to quantify the biomass of a species is of interest, a more precise shape approximation is needed (Chapter 7.1). Therefore, the allowed spatial uncertainty has to be chosen with care as reducing the spatial uncertainty can only be achieved by performing a re-annotation. For such a re-annotation task prior location information can be used, reducing the time needed. In an SPC task, a set of OOI, e.g. collected in a prior annotation task, is investigated and labeled. This task is much faster than the WIS task and can be considered as a "small scale FIA". As for the FIA task only the class label needs to be selected and no new region information is collected. SPC is only used for manual evaluation processes of automated systems in this thesis.

3.2   Quality assessment

Annotation correctness and reproducibility can be measured to assess the quality of an annotation (AQ challenge). The correctness evaluation requires the existence of a ground truth, which for instance can be estimated by computing a "gold standard". The reproducibility can be estimated by a re-annotation task by the same observer (intra observer agreement) or by

a different observer (inter observer agreement).

3.2.1 Gold standard

Computing a "gold standard" is often essential as no ground truth exists. A gold standard allows to agree on a class $\omega^{(i)}$ and a region $\bar{\mathbf{r}}^{(i)}$ for an annotation:

$$g^{(j)} = (\bar{\mathbf{r}}^{(j)}, \omega^{(j)}).$$

This agreement can either be made by just one expert observer (single expert) or as a consensus of several export observers (consensus expert).

In the single expert method the expert observer performs the annotation task of an image-set and the results form the gold standard. Applying the single expert method, therefore needs an expert observer which has extraordinary annotation skills and expertise. In order to assess the quality of this gold standard, the observers' intra-observer agreement should be computed (see Paragraph 3.2.3).

In the consensus expert method $|O|$ different observers perform the same annotation task independently on the same image-set, using the same annotation software, the same classes and the same annotation protocol. The protocol includes an agreed upon set of region descriptors, which can also be class specific as illustrated in Figure 3.1 allowing different degrees of special annotation uncertainties (AU challenge). At this stage no interaction about the annotation task between the observers should be allowed. It might be necessary to conduct a training session previously to agree on annotation classes and annotation protocol (Section 6.1.3).

Different observers' annotations from the same image I_t are then compared. Overlapping annotations $a^{(i)}, a^{(i')}, ...$ (with e.g. |region overlap| $\geq \frac{2}{3}$) with matching class labels $\omega^{(i)} = \omega^{(i')} = ...$ from at least $\kappa < |O|$ different observers form the basis for a gold standard annotation. The region $\bar{\mathbf{r}}^{(j)}$ is defined by the single $\mathbf{r}^{(i)}, \mathbf{r}^{(i')}, ...$ e.g. using the intersection of all $\mathbf{r}^{(i)}, \mathbf{r}^{(i')}, ...$. It also has to be declared how to deal with non- trivial cases (see Figure 3.2 d) and e) for examples).

Sometimes physical ground truth data in form of samples can also be used to create a gold standard. The use of such methods can be problematic as for instance a box scorer might grab objects which are invisible in the corresponding image as they were covered by sediment during image acquisition. Also the relocation of the sample extraction can be problematic.

3.2.2 Correctness

Computing the correctness is similar to computing the performance of a classifier as described in Section 2.10. It can only be conducted, if a ground truth in form of a gold standard is available. As for the evaluation of a computer vision system, the type of ground truth and
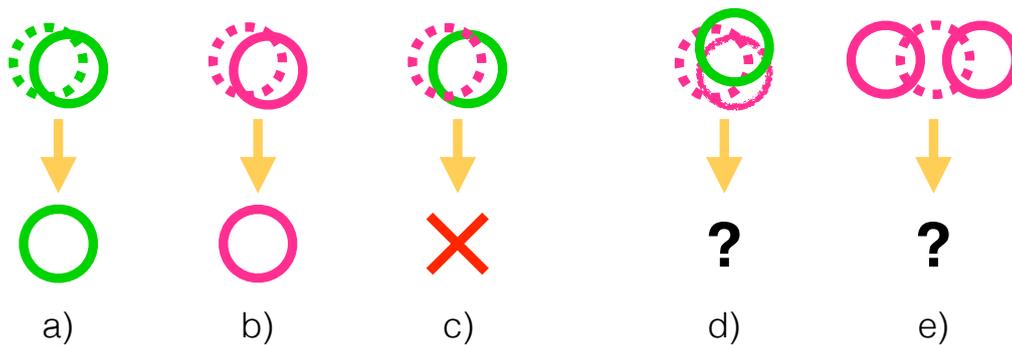
Figure 3.2: Examples for creating a gold standard. The top row shows the manual annotations, the bottom row the corresponding gold standard annotations. Different class label are symbolized by different colors and different observers are symbolized by different line types. a) and b) show the straightforward creation of a gold standard annotation with identical class labels for two different classes (a) class label "green" and b) with class label "magenta"). c) shows a class mismatch resulting in discarding of both annotations. d) and e) symbolize non-trivial cases which have to be handled with care. Discarding those annotations or creating a gold standard annotation are both reasonable options.

also the field of application needs to be considered. It has to be distinguished between WIS, FIA and SPC. The correctness of FIA and SPC runs can be computed in a straightforward as true positive (TP), true negatives (TN), false positives (TP) and false negatives (FN) can be defined. Similar to a classifier considering a two class problem with $\omega^{(i)} \in \{0, 1\}$, the annotation labels of the gold standard need to be compared to the new annotation. Matches are counted as TPs or TNs and mismatches as FPs or FNs, respectively. The correctness can then be computed as described in Section 2.10 using precision (eq. 2.3), recall (eq. 2.4), TNR / FPR (eq. 2.5) and ACC (eq. 2.4). For the WIS task the spatial uncertainty (AU challenge) needs to be considered. Similar to the computation of a gold standard, a perfect spatial match of two annotations can not be expected for most underwater image annotations. An allowed spatial uncertainty therefore needs to be defined which relies on the field of application and / or OOIs. It should be noticed that the ACC might not be the appropriate score for the WIS evaluation as the number of TNs might be greater by orders of magnitude than the number of positive annotations. This also depends on the annotation strategy and available ground truth.

The described methodology can also be extended to multi-class annotations ($|\Omega| > 2$), by computing confusion matrices and computing class-wise correctness scores as described in Section 2.10

3.2.3 Reproducibility

As mentioned earlier, the reproducibility is one corner stone in good scientific practice. The reproducibility of a manual annotation task is investigated by computing the inter- and intra-
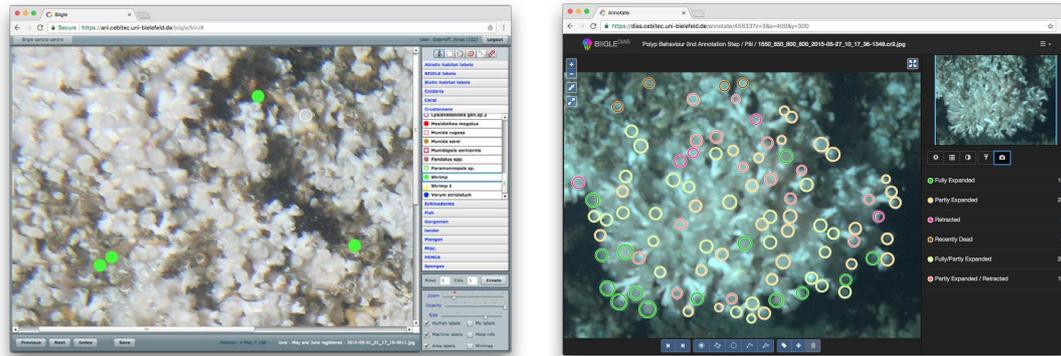
Figure 3.3: Screen-shots of the two BIIGLE systems. On the left a screen-shot of the old system (BIIGLE 1.2x) and on the right a screen-shot of the new system (BIIGLE 2.0).

observer agreement. The term "inter" is referring to the case where the observers agreement is investigated between two different observers o_j, o_k and the intra-agreement the case where the agreement is investigated between two different annotation runs an the same image data conducted at different time points by the same observer.

In the context of this thesis the accuracy (ACC) is used to compute both agreements. The annotations from one observer or one annotation run are used as the ground truth and are compared to the corresponding annotations from the other observer or annotation run. The ACC has to be interpreted with care and is not a suitable score for datasets where classes are highly unbalanced. In such cases precision (eq. 2.3), recall (eq. 2.4) and F_1-score (eq.2.6) are more suitable if applied class-wise.

3.3   BIIGLE

Large image sets require a) tools to optimize the process of manually annotating images and b) the automation of the annotation process. The manual annotations in this thesis were placed using the BIIGLE 1.2x (`https://ani.cebitec.uni-bielefeld.de/biigle/bin/`) (Benthic Image Indexing, Graphical Labeling and Exploration) software [Ontrup et al., 2009] or the BIIGLE 2.0 software (`www.biigle.de`) [Langenkämper et al., 2017], a new implementation of BIIGLE. Both software tools are Rich Internet Applications allowing users, e.g. biological experts, to browse and annotate underwater images from all over the world, as long as they have a device with Internet access (Figure 3.3. Easy and location independent access can be a key factor in many international collaboration projects.

BIIGLE 1.2x is an Adobe flash based system realized using the Flex 3 SDK. A flash player plugin is needed to run the software in a browser. This software was one of the first software tools running in a browser to manually annotate underwater images. With this software images can be browsed and annotated. For the annotation, first an annotation class ω has to

be selected. Next, the annotation shape has to be selected which can either be a point shape or a rectangle, the latter can be rotated after placing. The placing is carried out by clicking with a computer mouse on the position of the OOI in an image. Furthermore, BIIGLE 1.2x allows to measure OOIs in the images, if pixel size ratios are available (e.g. by the presence of a laser-points). In BIIGLE 1.2x a broad spectrum of tools is implemented for the information visualization based exploration of the images and their annotations [Schoening et al., 2009]. With this tool-set "link and brush" techniques allow to select images containing a specific number of annotations of a specific annotation class.

The BIIGLE 2.0 software is realized without the Adobe flex SDK, so no additional plugins are needed to run this software in a browser. The workflow for the manual annotation is similar. After selecting the image for annotation, first the annotation class is selected and second the annotation shape is selected. Depending on the selected shape the annotation is then placed by multiple mouse-clicks on the image. The variety of shapes compared to the old BIIGLE is higher and includes points, rectangles, circles, lines and polygons. Despite many improvements of the BIIGLE 2.0 software, the ability to enable the user to flip quickly between images was one of the most important enhancements for the purpose of this thesis. This feature made the identification of the different gradients of polyp activity possible (Section 6.1.3) by allowing the human observers to learn the different characteristics for different levels of polyp activity.

4 Underwater imaging

In the following section, some underwater image acquisition platforms are briefly discussed followed by a more detailed description of fixed underwater observatories, as the majority of the images analyzed in this thesis were recorded by such a platform. In general, underwater imaging is affected by the image acquisition challenges (OP challenge and DOP challenge), but the distinctiveness of these affects can be reduced during image recording. For example in order to reduce the backscattering, the distance between illumination source and camera needs to be maximized [Funk et al., 1972]. Of course this maximization has its limits as it comes with the cost of illumination loss. This is not discussed further here as the focus of this thesis is on the processing of the recorded images. Additionally, image systems are often calibrated in order to for instance compensate camera-lens distortion and to be able to measure sizes of OOIs in the recorded image. This is not applied here, as the provided image material are time-series images, imaging a fixed location and changes over time in form of for instance species distribution can be identified without such calibrations. Nevertheless, it needs to be considered in the interpretation of the results as images patches towards the boarder have a different pixel to area ratio than patches in the center of the images.

4.1 Digital imaging

Nowadays, most photos are recorded digitally and a therefore referred to as digital images. Images are recored by digital cameras which are either charge coupled devices (CCD) or complementary metal oxide silicon (CMOS) devices [Nixon et al., 2012a]. In CCD and CMOS cameras the light traveling to the camera chip is first passing a micro-lens array followed by color filter array. Color filters are typically arranged according to the "Bayer" pattern [Nakamura, 2005; Bayer, 1976], with twice as many green filters as blue or red filters [Nakamura, 2005]. Sensor measurements need to be transfered to a "normal" RGB image, which is called RAW processing. This includes demosaicing of the color filter array and may also include white balancing (see Section 5.2.1), lens distortion correction and noise reduction. The raw processing furthermore allows to produce RGB images with a high dynamic range. Special cameras like infrared cameras or hyperspectral cameras with more sensing bands [Nixon et al., 2012a] also exist and also have been used for underwater imaging. Hyperspectral imaging has been deployed successfully to create seafloor maps [Johnsen et al., 2016]. A modified camera, where an infrared filter is removed from the camera lens application to capture a broader light-spectrum has been successfully applied to improve automated detection of benthic images [Beijbom et al., 2016].

4.2 Underwater in-situ imaging

Underwater imaging has a long tradition in marine sciences as it allows the in-situ exploration and visualization of underwater environments [Kocak et al., 2005a]. Underwater imaging

can in principle refer to video or still image recording. Techniques applied in this thesis concentrate on still image processing, but can easily be applied to video data by extracting video frames as images.

A variety of platforms exist to record images underwater. A very common platform, at least for deep water industries, are remotely operated (underwater) vehicles (ROVs). ROVs are tethered vehicles operated by a team of experts which are located on the connected vessel (ship). In addition to sensors, ROVs are usually also equipped with tools and manipulator arms for underwater construction. ROVs often allow live video recording which is also used for navigation.

In recent years automated underwater vehicles (AUVs) have increasingly be deployed. AUVs can be used for imaging the seafloor over large areas automatically. Usually diving plans are programmed before deployment and recorded images are transfered after recovering the AUV. More details of AUVs can for instance be found in [Wynn et al., 2014].

More traditional image recording platforms are towed camera systems like the ocean floor observatories system (OFOS). A sledge mounted camera is towed behind a vessel. The distance between ground and OFOS is controlled by a winch operator. ROVs, AUVs and towed platforms, can be used to image the sea-floor or water column. The sea-floor is typically imaged with a down-facing camera either by creating mosaics by stitching overlapping images (e.g. [Singh et al., 2004]), which also can be extracted from a video (e.g. [Nicosevici et al., 2009]), or by image sampling with no overlap (e.g. [Schoening et al., 2012; Schoening et al., 2016a]).

A different type of platform are fixed underwater observatories (FUOs), also known as fixed point [Cristini et al., 2016] or Eulerian observatories [Lampitt et al., 2010]. These multiple sensor platforms, are stationary sea-floor observatories deployed for long-term monitoring of a small area of interest. Most images considered in this thesis were recorded by such a camera equipped fixed underwater observatory and FUOs are therefore discussed in detail in the following section.

4.2.1   Fixed underwater observatories

Fixed underwater observatories (FUOs) are typically equipped with a variety of sensors measuring water properties like, Chloropyhl concentration, salinity or temperature. Some FUOs are also able to collect samples of water, biota or particles [Lampitt et al., 2010]. Here, only FUOs equipped with camera systems are considered. FUOs constitute a rich collection of quantitative and qualitative data for high temporal resolution monitoring of smaller areas of interest over longer time periods with a high temporal resolution. In particular, camera equipped FUOs provide a unique opportunity to gain general knowledge about the present species' natural variations related to behavior, describing their spatial and temporal distribution in the various microhabitats covered by the camera frame of approximately $1\ \mathrm{m}^2$ to $20\ \mathrm{m}^2$ over time. Information of such natural variation is essential for detecting long-term changes, seasonal fluctuations, sudden events, or possible impact from anthropogenic activities such as oil drilling activities [Nilssen et al., 2015].

In recent years the number of FUOs equipped with cameras has increased a lot [Aguzzi et al., 2011]. The NEPTUN Canada observatory system [Barnes et al., 2015; Ocean Network Canada, 2014], for instance, located in the north-east Pacific Ocean off the coast of British Columbia, includes several FUOs. The system became first operational in 2009. The NEPTUNE FUOs are located in a coastal area ($20-100$ m), on the continental slope ($400-1000$ m) and deep sea area (2300 m) and are all connected to main land for power-supply and communication purposes. Another example for a coastal FUO is OBSEA [Aguzzi et al., 2011], which is located 4 km off coast at a depth of only 20 m in the Mediterranean Sea. It was installed in 2009 and its camera is recording only during daylight hours due to the absence of an artificial light source. The ALOHA cabled observatory [Howe et al., 2015] is located in the North Pacific, 100 km north of Oahu, Hawaii. It is installed at a depth of about 4700 m in the deep ocean. A camera was installed in 2011. As no sunlight reaches such depths its camera system is equipped with artificial lights. DELOS [Vardaro et al., 2013] is another observatory system consisting of two FUOs. It was first installed in 2009 at a depth of 1400 m. The FUOs are located in the Southeast Atlantic with about 16 km distance between them. Similar camera systems are installed on both FUOs consisting of an artificial light source and two cameras each, one wide-angle camera and one with a closer view. Other FUOs, including FUOs without cameras are managed in the fixed point open ocean observatory network (FixO3) [Fixed-Point Open Ocean Observatories, 2015].

Lofoten-Vestarelen observatory Most digital images processed in this thesis were recorded by the FUO LoVe (Lofoten -Vesterålen), located in the north of Norway 22 km off the coast at a depth of 260 m (N $68°54.472'$, E $014°23.078'$) [Godø et al., 2014]. It was first deployed in September 2013 and is connected to main land for power-supply and communication. It is equipped with different sensors, for measuring water conditions, including Chlorophyl concentration, salinity, conductivity, temperature, current speed and direction. LoVe is equipped with a digital camera, a Canon EOS 550D in a waterproof housing (METAS DSF5210) with a flash (METAS DSF4365) for artificial illumination due to the absence of daylight at this water depth. All data recorded by this FUO is publicly accessible via the LoVe web-page: `http://love.statoil.com`. The camera is mounted on an adjustable beam allowing to pan and tilt the camera. The camera was oriented to face a *Lophelia pertusa* reef with a camera angle of $45°$ during image recordings which are processed in this thesis. Still images were recorded hourly. The image frame was selected to cover the different micro-habitats at the coral reef [Mortensen et al., 2006; Purser et al., 2013; Jonsson et al., 2004] consisting of live corals, areas with dead fragments and coral rubble where small coral fragments are mixed with sand. To get a first impression of the LoVe images, two of the recorded images are presented in Figure 4.1.

The camera platform was re-deployed after a maintenance break from mid-June until the beginning of October 2014. The location of the camera platform moved a little further away from the coral-reef after the re-deployment. During this re-deployment a base for the camera satellite was installed. Retrieving and re-deploying the camera system is now possible without changing the position. The maintenance break could be used to clean the housing covering
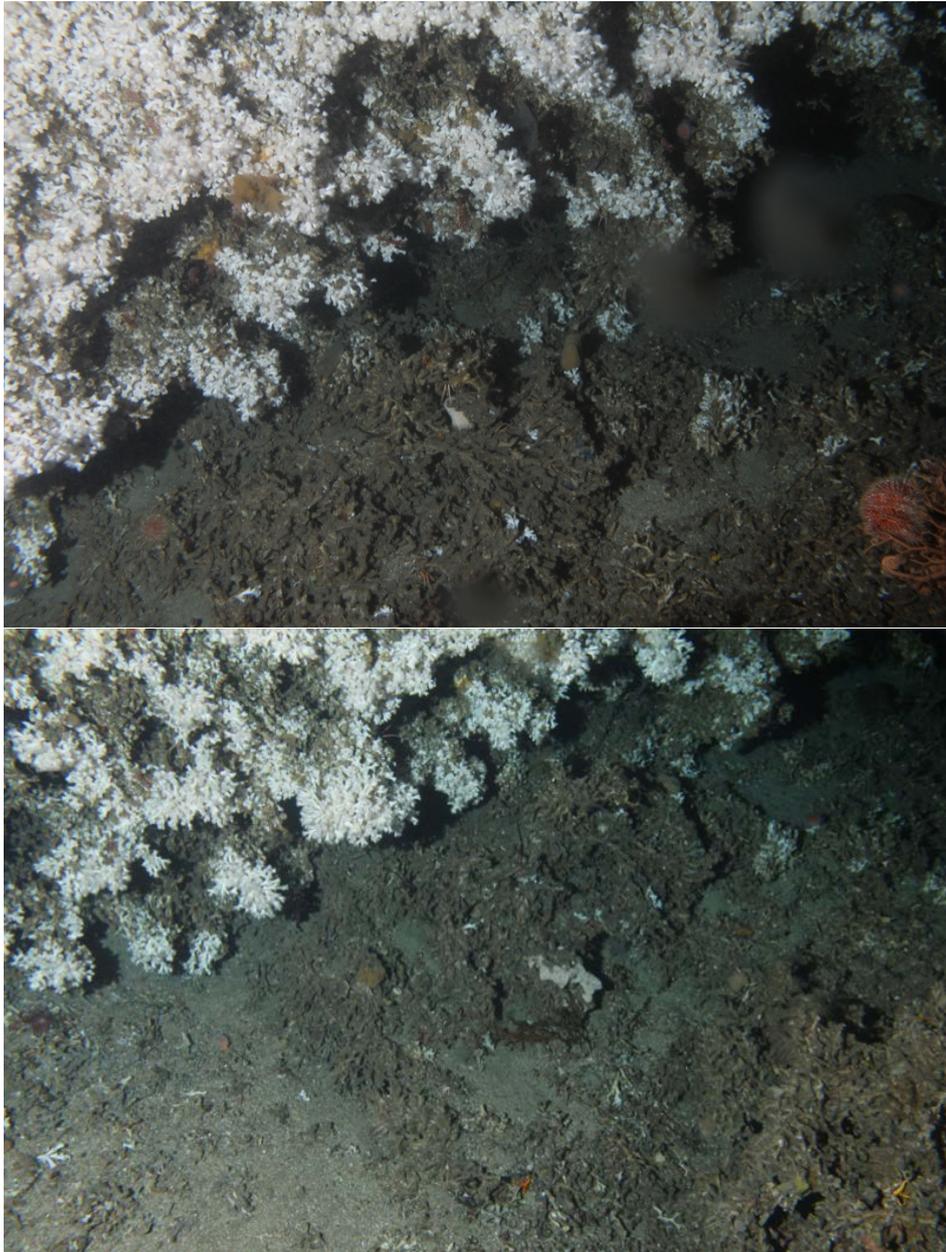
Figure 4.1: Two LoVe images recorded during the different deployments. At the top an unmodified image collected at the 21st of April 2014 6:17:19 during the first LoVe deployment. At the buttom an unmodified image collected at the 3rd of April 2015 7:59:31 during the second LoVe deployment. For both, the camera is facing towards the same *Lophelia pertusa* coral reef block. In the upper left parts of the images mostly live corals can be found, in contrast to the lower right parts where mostly dead corals fragments are visible. A coral rubble area can be found in between, where small coral fragments are mixed with sand. Comparing both images, it can be noticed that the camera platform moved further away from the coral reef for the second deployment.

the camera lens, as it showed biofouling on two fingerprints which were placed accidentally during the first deployment. Furthermore, the camera setup was changed allowing to record raw images with 14 bit per color channel instead of only 8 bit in the previous setup. Raw image recording not only increased the channel depth, but allows to apply a post white-balancing of the digital images with less information loss. Due to technical challenges, several sensors from LoVe are off-line since November 2015, including the camera.

In this thesis, two different image sets from the LoVe observatory are analyzed (see Figure 4.1 for examples). The first set (LoVe image set 1) contains 1140 JPG images (8 bit per color channel) from the first deployment recorded within 6 weeks from 1st of May 2014 0:17:19 until 18th of June 2014 08:17:19. The second set (LoVe image set 2) contains 4862 RAW (14 bit per color channel) images recorded after the re-deployment from 3rd of April 2015 7:59 until 10th of November 2015 13:17. For both image sets, images were recorded hourly and have a resolution of 5184×2456 pixels. The distance between camera and coral reef was about $2 - 3$ m [Godø et al., 2014] and the imaged area is about 10 m^2, resulting in images with about 2×10^6 $\frac{\text{pixels}}{\text{m}^2}$. Due to the change of position for the re-deployment, the imaged area is a little smaller for the first deployment than for the second deployment, resulting in a slightly higher $\frac{\text{pixels}}{\text{m}^2}$ ratio for the first deployment.

4.3 Underwater wet-lab imaging

Another type of images analyzed in this thesis are wet-laboratory images. In contrast to underwater imaging, wet-lab imaging can be conducted in a highly controlled environment. Experimental samples can either be imaged without removing from the experiment, or by extracting for image acquisition purposes. The latter allows to maximize the stability of the imaging conditions and of course makes the **DOP challenge** obsolete. The effect of the **OP challenge** also is reduced as variance in object-camera distance can be minimized assuming similar sizes and structures of samples in the experiment. Highly stable imaging conditions also increase the reproducibility of the results of the experiments, which sometimes has become a problem in science [Baker, 2016].

The images of the wet-lab study analyzed in this thesis, were recorded with an affordable camera setup. Samples were removed from the experimental environment and cleaned for imaging. To eliminate light refractions and reflections samples and camera, the latter protected by a waterproof housing, were placed underwater inside a (water filled) cylinder. The camera, a Canon PowerShot G10, was placed on a tripod (or quadpod as it had 4 legs) to keep the distance between camera and samples stable (Figure 4.2 on the left). The camera was set in automatic mode with macro function and internal flash on. In order to be able to compensate possible illumination variation, e.g. due to presence of varying daylight, change of flash intensity or the automated white-balancing of the camera, a color reference was placed near the imaged sample (Figure 4.2 on the right). In the wet-laboratory study the task was not only to estimate the presence or size of specific species, i.e. the calcareous alga sample, but to investigate possible treatment related color change. The reference plate is therefore essential for a comparative study of the color over time, in the described setup.
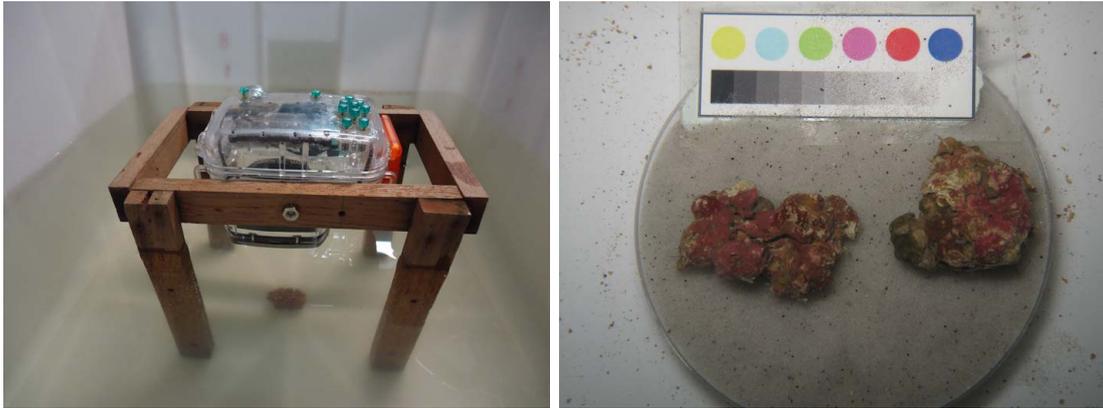
Figure 4.2: The wet-lab imaging setup and a wet-lab image. On the left, the camera setup for the wet-laboratory allows to keep the distance between camera and imaged objects stable. Images were taken underwater to eliminate light refractions and reflections. The camera, not included as it was used for recoding the digital image, is placed in a waterproof housing. On the right one image, recorded during the experiment, shows two calcareous algae covered rodoliths below the color reference plate.

The digital images of the samples analyzed in this thesis, were taken during a weekly examination. The images have a resolution of 3456×2592 pixels. Each week 63 images, displaying 2 samples, were taken summing up to 630 images for the whole experiment. More details about the experiment are given in Chapter 7.1.

5 Preprocessing for underwater images

The follwoing section is mainly based on publication 1 and 7. Extent and notation have been adjusted for the need of this thesis.

Underwater computer vision often involves preprocessing of images to improve the visual quality, by reducing the effects of image acquisition challenges (OP challenge and DOP challenge). Here, it is called preprocessing as it is applied prior to the computer vision methods. In general, image preprocessing can also include image transformations to compensate camera-lens distortion and size normalization. In this thesis the latter processings were not applied and image preprocessing refers to the compensation of color shifts, illumination correction and general image improvements like contrast enhancement or sharpness corrections.

Image preprocessing by manual tuning of individual images should be avoided, at least for the computer vision task, as it may be affected by each individual observer and settings including monitor and workplace illumination with varying natural and / or artificial illumination affecting color perception. An individual manual tuning is also not applicable with increasing number of images. For computer vision the main aim of the preprocessing is not to please the visual appearance or to compute accurate colors, but to increase the color constancy over all analyzed images. The variation of the appearance of OOIs with the same visual spectrum need to be reduced to a minimum while the differences between different OOIs, e.g. different species, need to be enhanced.

In the following, a brief overview about methods for preprocessing of underwater images is given. Methods applied in this thesis are described in more detail in Section 5.2.

5.1 Overview of underwater image preprocessing methods

Underwater image processing has been a very active research field in recent years. Reviews of different methods can be found in [Kocak et al., 2005b], [Kocak et al., 2008], [Schettini et al., 2010], and [Lu et al., 2017]. Methods are either designed to process underwater images recorded under natural or artificial illumination, with a specific camera orientation (vertical vs. oblique) or in general without specific restrictions. The presence of natural light has a non linear influence on the color shifts. In such shallow water settings, the appearance of objects is not only dependent on the distance between camera and object but also on the distance to the water surface. Furthermore, natural illumination is affected by surface waves, refraction is influenced by water wave turbulences, resulting in sunlight caustics. The use of artificially illumination is often causing an illumination cone visible in the image. Illumination of objects is then also depending on the distance to the center of the illumination cone. Camera orientation typically has an influence on the variation of distances between camera and imaged objects. An oblique camera orientation typically shows higher variation of distances, which result in higher variation in color shifts in the images, than a vertical camera orientation setup.

In order to correct illumination variation, visible for instance as sunlight caustics in shallow water images or as an illumination cone in artificial illuminated images, typically local image corrections are applied. Examples are local histogram equalization methods (Section 5.2.3) such as CLAHE [Zuiderveld, 1994] (Section 5.2.4) or ACE [Rizzi et al., 2002] (Section 5.2.5). Other methods model the illumination cone as a Gaussian distribution [Garcia et al., 2002], a two-dimensional polynomial spline [Rzhanov et al., 2000] or a 2D polynomial function, as in fSpice [Schoening et al., 2012] (Section 5.2.10) or in [Singh et al., 2007]. One method for the explicit removal of sunlight caustics visible as wave-caused illumination patterns like sun flickering is presented in [Gracias et al., 2008]. Multiple images of the same spot e.g. extracted from overlapping video frames are used to compute a median intensity image. Such a computed median image then provides the information needed to compensate such an effect. A similar approach is presented in ["A new method for ecological surveying of the abyss using autonomous underwater vehicle photography"]. Again a median image is computed to correct for illumination fluctuations.

Color shifts are typically corrected by balancing the color distribution. This is often achieved using the gray world assumption [Buchsbaum, 1980] or similar approaches like histogram equalization (Section 5.2.3) or histogram stretching [Iqbal et al., 2010]. In [Beijbom et al., 2012] for instance image colors corrected by stretching each color channel (RGB) individually, equalizing the color shift only. This method is therefore also utilizing the gray-world assumption. In [Petit et al., 2009], color shifts towards blue colors are removed by modeling the light attenuation using quaternations. Water colors are shifted to gray colors while other colors are shifted to balance the color distribution. Another method using a modification of the gray-world assumption is presented in [Ancuti et al., 2018]. The gray-world assumption is modified that each channel is modified individually accounting for the different grades of attenuation. Other methods exists which correct the color shifts in similar ways e.g. [Erat et al., 2017] or [Ghani et al., 2017]. The latter is using an adaption of the CLAHE method applying histogram clipping and equalization to overlapping image regions.

Another correction target is the removal of "haze" occurring in turbid water. It is caused by high scatter-rates due to an increased number of particles in the water. One method focusing in the de-hazing is presented in [Trucco et al., 2006]. The method is based on a simplified Jaffe-McGlamery model [Jaffe, 1990] for natural illumination only and is therefore optimized for shallow water images. The adopted model considers forward-scatter only, simplifying the Jaffe-McGlamery model to filtering the image in the frequency domain. In [Carlevaris-Bianco et al., 2010] an adaption of the image transmission model by Duntley [Duntley et al., 1957] is used to de-haze underwater images. Model parameters are estimated directly from the images. The scene depth is estimated utilizing the wavelength dependent attenuation of the light. The difference between stronger attenuated light e.g red and lighter attenuated light e.g. blue provides the information to calculate the scene depth. Further parameters are "airlight", estimated by the pixel value furthest away from the camera and scene radiance modeled by a Markov random field under white Gaussian noise.
Scattering visible as haze can also be reduced with polarization filters. In [Schechner et al., 2005] for example the used camera is equipped with a circular polarization filter. Two

images are taken with different filter orientations to record one image with minimal scatter and one with maximal scatter. A difference image of both is then providing the information to compute an image without haze. This method was developed further in [Treibitz et al., 2009]. In the enhanced method the imaged scene is also illuminated with a polarized artificial illumination, increasing the difference between the maximal and minimal backscatter images. In general, de-hazing can also be performed by increasing the image sharpness in form of for example anisotropic and wavelet filtering [Arnold-Bos et al., 2005; Bazeille et al., 2006] (Section 5.2.9).

In recent years, stereo-camera systems became quite popular. Such systems allow to computer the object-camera-distances on a pixel level from stereo-image-pairs. On method explicitly designed for stereo-camera-systems is presented in [Roser et al., 2014] allowing to compute object-camera distances on a pixel-level to correct both for color and illumination with respect to the computed object-camera distances. Other stereo-camera optimized methods are presented in [Mahon et al., 2011] and [Bryson et al., 2012]. The first one estimates parameters for a polynomial vignetting model from the stereo image pair to remove an vignetting effect. Images were taken either shortly after sunrise or before sunset to avoid sun-flickering effects caused by waves. The imaged scene was additionally illuminated with an artificial light source (LED). The second one is applying the gray-world assumption to pixels with similar object-camera-distances over several images. The object-camera-distances are again computed from pairs of stereo-images.

Other methods are estimating the camera-object distances from single images. The method presented in [Chiang et al., 2012], for example, is estimating the camera-object-distances using the dark channel prior method [He et al., 2011]. Haze, caused by scattering of light on small particles, is removed by subtracting a wavelength and water-type depended coefficient. The coefficient is then weighted by the estimated camera-object-distances on a pixel level. Another method estimating the camera-object distances from single images is presented in [Barros et al., 2018].

Recently also ML algorithms have been used to learn the true color of an image. The method proposed in [Barron, 2016], for instance, was successfully applied to boost classification performance on standard image benchmarks. Another ML based preprocessing is presented in [Shen-Chuan Tai, 2017] using a Random Forest classifier to compute object-camera-Distances on a pixel-level.

The optimal solution of course would be to know the correct color of the imaged objects. The use of color references for instance in form of a color plate would come quite close to this, but is most often not applicable for *in-situ* imaging. It is therefore used only rarely for preprocessing the images, e.g. by [Åhlén et al., 2007].

In general, preprocessing for color and illumination correction benefits from using cameras with an increased dynamic range (more than 8bit per color channel) as it enables to reconstruct colors even in bright or dark areas. Digital images recorded in RAW mode often feature higher color depths and also allow to modify the white balancing (WB) after recording with reduced information loss.

5.2   Applied preprocessing methods

Only a selection of preprocessing methods were applied in this thesis. Methods were chosen due to the high image quality and given camera features, excluding methods designed for instance developed for stereo-cameras or polarization filters. The selected methods are described in the following and exemplary results of applying those methods are presented in Figure 5.1.

5.2.1   Auto white balancing

Several algorithms exist for white balancing (WB). In general all algorithms apply individual transformations for each color channel individually to *balance* the colors in the image. WB is applied globally so no illumination fluctuations within an image are corrected. One of the simplest methods for WB is the White Patch Retinex (WPR) algorithm. The brightest pixel in an image defines the *white point* and is used to scale all color channels individually to maximum range. In order to increase the robustness, the top p-percent of the pixel values are ignored [Ebner, 2007].

Another WB method is based on the gray world assumption [Buchsbaum, 1980]. Color channels are scaled such that the average of each channel is set to the center of the color scale, e.g. for an 8bit channel:

$$\sum_{x,y}^{w,h} \mathbf{I}(x, y, c) = 127 \; \forall c \in \{r^{(\text{RGB})}, g^{(\text{RGB})}, b^{(\text{RGB})}\}.$$

In the dcraw software [Coffin, 2016] this scaling is limited so that only 1% of the pixel-values are clipped [Ebner, 2007].

5.2.2   Fixed white balancing

In contrast to the auto WB, parameters for the scaling of each color channel is computed on a set of images $\{I_t\}_{t=1,\dots,N}$ for fixed white balancing (FWB). The average scaling parameters are then applied to all images $\mathbf{I}$. This FWB can be applied for images recorded under similar conditions. Camera object distances need to be stable as for images recorded with a fixed underwater observatory (Section 4.2.1). FWB allows to produce natural colors now with an increased color constancy over all $\mathbf{I}$.

5.2.3   Histogram equalization

Histogram equalization (HE) is usually applied to increase the contrast of a gray image, but it can be applied to color images in a straight-forward way to stretch the intensity range.

Each pixel $\mathbf{I}_t(x, y)$ is transformed by

$$\mathbf{I}_t^{(HE)}(x, y) = \frac{(L-1)}{w \cdot h} \sum_{j=0}^{\mathbf{I}_t(x,y)} n_j$$

with $\mathbf{I}_t$ in range $[0, L-1]$, e.g. $L = 256$ for 8-bit images, and n_j the number of pixels with intensity j [Gonzalez et al., 2008].

5.2.4 Contrast limited adaptive histogram equalization

Contrast limited adaptive histogram equalization (CLAHE) is based on the HE method. HE is here applied on local image regions and and balanced regions are then fused using a bilinear interpolation scheme to avoid visible region boundaries [Pizer et al., 1987; Zuiderveld, 1994]. Furthermore, the HE balancing is limited. During histogram creation pixel value occurrences are clipped and redistributed equally over the whole histogram. Pixel values with a high occurrence are therefore shifted less than in AHE and the equalization is limited.

5.2.5 Automatic color equalization

Automatic color equalization (ACE) [Rizzi et al., 2002; Rizzi et al., 2003] was not particularly designed for preprocessing underwater images. Nevertheless, it could successfully be applied to increase the accuracy of an automated fish classification system [Chambah et al., 2003]. It is a perceptually inspired method combining the "Gray World" and "White Patch" color scaling methods [Rizzi et al., 2003]. First, for each pixel $\mathbf{I}_t(x, y)$, differences of its neighboring pixels are applied to a saturation function. These differences are weighted by their distance to the pixel $\mathbf{I}_t(x, y)$ and summed up. Second, new pixel values are scaled using a mixture of white patch and gray world scaling. The original ACE implementation is computationally expensive, an efficient implementation [Getreuer, 2012] was used in this thesis. As it computes the color correction locally it also corrects for illumination fluctuations.

5.2.6 Local space average color scaling

The local space average color scaling (LSACS) [Ebner, 2007] models a non- uniform illumination using a Gaussian kernel with respect to the gray world assumption [Buchsbaum, 1980]. The color-corrected image is computed by a pixel- and channel-wise devision of the original image by two times the Gaussian filter image.

5.2.7 White Patch Retinex and Gray World combination

The White Patch Retinex and Gray World combination (WPRGW) algorithm combines both auto WB assumptions: the white patch retinex and gray world assumption. It also models

a non- uniform illumination using a Gaussian kernel [Moore et al., 1991; Ebner, 2007]. The color corrected image is obtained by a pixel- and channel-wise subtraction of the Gaussian filter image from the original image, followed by a rescaling.

5.2.8   Unsupervised color correction

The unsupervised color correction (UCC) method can be divided into three processing steps [Iqbal et al., 2010]. In the first step the RGB color channels are equalized as in the auto WB method. In the next step, a histogram clipping and stretching is applied for each color channel individually using upper and/or lower limits (balanced or unbalanced clipping) dependent on the dominant color channel. Finally, an additional histogram clipping and stretching in the HSI color space is applied (balanced clipping). This final stretching is only carried out for the $s^{(\text{HSI})}$ and $\mathbf{I}^{(\text{HSI})}$-channels, but again for each one individually.

5.2.9   Automated underwater image preprocessing

The automated underwater image preprocessing (AUIP) method comprises different processing steps [Bazeille et al., 2006]. First, a potential moiré effect is removed. Then, a homomorphic filtering, a wavelet denoising and an anisotropic filtering is applied in the $Y'C_BC_R$ color space (Section 2.2.6). Finally the mean color of the image is equalized in the RGB color space. A C++ adaption of the provided Matlab code was used to compute the AUIP on the images.

5.2.10   Feature space based illumination and color enhancement

In the feature space based illumination and color enhancement (fSpICE) preprocessing method a Gaussian filtering is applied to correct the variation of illumination [Schoening et al., 2012]. The best kernel size for the filtering is determined by evaluating distributions of manual annotations of different classes in a chosen feature space with cluster indices similar to the preprocessing ranking described in Section 5.3.1. The kernel size is optimized for each image set individually. The pixel-wise difference between the original and the Gaussian filtered image forms the illumination-corrected image. Finally, to equalize the brightness of all images in one image set, a gamma correction is applied for each image and each RGB channel individually shifting the abundance maximum (histogram peak) of the corresponding color histogram into middle range.

5.2.11 $l\alpha\beta$ underwater color correction

The $l\alpha\beta$ underwater color correction method [Bianco et al., 2015] corrects the color shift of underwater images. The method was originally applied to correct color shifts caused by

imaging with colored illumination [Reinhard et al., 2001]. Colors are balanced in the Ruderman component color space $l\alpha\beta$ [Ruderman et al., 1998] (Section 2.2.5). Images are transfered to this color space and the median of the α and β channels (color channels) are subtracted from the corresponding α and β channels to shift the median of both color channels to the "white point" ($\alpha = 0, \beta = 0$). An illumination correction can be performed additionally on the l-channel using for instance CLAHE (see above).

5.3   Assessment

Quantifying the quality of image preprocessing methods for a set of images can be challenging. A straightforward method, of course would be to have a ground truth for instance in form of a color reference plate. Differences between colors of the plate imaged in air and reconstructed colors give an objective assessment as done by [Åhlén et al., 2007]. Usually there exists no real ground truth or a reference object/signal in the images to evaluate the quality of the preprocessing.

Several methods have been published that use a visually subjective quantification as for instance [Garcia et al., 2002; "A new method for ecological surveying of the abyss using autonomous underwater vehicle photography"]. Image properties, like colorfullness, sharpness and contrast [Yang et al., 2015; Panetta et al., 2016] or image blur [Trucco et al., 2006], range of visibility [Schechner et al., 2005] are used to estimated the performance of preprocessing methods objectively.

In a computer vision application it is also possible to compare classification performance on different methods [Chambah et al., 2003] or to apply a cluster index ranking. The latter was developed within the scope of this thesis (contribution 1) and is explained in more detail in the following section.

5.3.1   Cluster index ranking

As stated earlier in this chapter, the variation of the appearance of OOIs with the same visual spectrum need to be reduced to a minimum while the differences between different OOIs, e.g. different species, need to be enhanced. Cluster indices were originally proposed to measure the quality of a clustering result [Caliński et al., 1974; Maulik et al., 2002; Davies et al., 1979]. Here, they are used to rank the results of different preprocessing methods. In contrast to their primary area of application, a heuristic tuning of cluster algorithm parameters to identify an optimal clustering, they are used to rank preprocessing algorithms with a given clustering. This clustering is obtained by manually annotating a set of OOIs. The score of a cluster index then indicates the discriminability of different annotation classes for the OOIs in relation to the compactness of a cluster of visually similar OOIs with the same annotation class ω in a chosen feature space. The increased discriminability might also improve the ability to localize and identify of OOIs by the human visual system.

For the purpose of ranking different preprocessing methods, manual annotations from a set of images are used. Instead of clustering a set of features, instances of each annotation class ω form a cluster in a chosen feature space. Hence, features like color histograms (Section 2.3.2) are extracted for each annotation as described in Section 2.3. If applied for choosing the best preprocessing for a computer vision approach, it is suggested to use the same feature descriptors for the ranking and the computer vision task. As it is not clearly defined what a good clustering result looks like, three different indices are applied to rank the different preprocessing methods. The cluster indices used in this thesis are:

Calinski Harabasz (CH)   The CH index measures the scatter of the cluster centers in relation to the scatter within the clusters [Caliński et al., 1974].

$$\chi^{\mathsf{CH}}(K) = \frac{\sum_{k=1}^{K} |C_k| d(\bar{\mathbf{v}}_k, \bar{\mathbf{v}})^2 / (K-1)}{\sum_{k=1}^{K} \sum_{\mathbf{v} \in C_k} d(\mathbf{v}, \bar{\mathbf{v}}_k)^2 / (L-K)}$$

where K is the number of clusters ($K = |\Omega|$), $\bar{\mathbf{v}}$ is the mean vector of all feature vectors, $\bar{\mathbf{v}}_k$ is the mean of the feature vectors in cluster k, $|C_k|$ is the number of vectors in cluster k and L is the total number of vectors and annotations, respectively.

Index-$\mathcal{I}$ ($\mathcal{I}$)   Similar to the CH, the index $\mathcal{I}$ uses the scatter within the clusters, but it is weighted reciprocally by the overall scatter E_1 [Maulik et al., 2002].

$$\chi^{\mathcal{I}}(K) = \left(\frac{1}{K} \times \frac{E_1}{E_K} \times D_K \right)^v \text{ with } K \text{ as above,}$$

$$E_K = \sum_{k=1}^{K} \sum_{\mathbf{v} \in C_k} d(\mathbf{v}, \bar{\mathbf{v}}_k) \ , \ E_1 = \sum_{i=1}^{L} d(\mathbf{v}^{(i)}, \bar{\mathbf{v}})$$

and D_k the maximal pairwise cluster distance $D_K = \max\limits_{i,j=1}^{K} d(\bar{\mathbf{v}}_i, \bar{\mathbf{v}}_j)$.

Davies Bouldin (DB)   The DB index uses the within cluster scatter (WCS_k) to score the distribution of the clusters [Davies et al., 1979]:

$$\mathsf{WCS}_k = \frac{1}{|C_k|} \sum_{\mathbf{v} \in C_k} d(\mathbf{v}, \bar{\mathbf{v}}_k) \ .$$

For each cluster k the pairwise maximum of the sum of the variance of cluster k and k' weighed by the distance of the two corresponding cluster centers is computed as $R_k = \max\limits_{1 \leq k' \leq K, k' \neq k} \left\{ \frac{\mathsf{WCS}_k + \mathsf{WCS}_{k'}}{d_{k,k'}} \right\}$. The sum of those maximums gives the score for the index:

$$\chi^{\mathsf{DB}}(K) = \frac{1}{K} \sum_{k=1}^{K} R_k \ .$$

Cluster Ranking In order to compute one score and to compare different cluster index results χ^*, with $* = \{\mathrm{CH}, \mathcal{I}, \mathrm{DB}\}$, first, a normalization to the range (0,1) of the results for the N different preprocessing for each index is required:

$$\hat{\chi}_n^* = \frac{\chi_n^* - \min\limits_{i=1..N} \chi_i^*}{\max\limits_{i=1..N} \chi_i^* - \min\limits_{i=1..N} \chi_i^*} \ .$$

Finally, the harmonic mean (H) of the three cluster indices is computed to rank the different preprocessing methods:

$$H = 3 \times \frac{\hat{\chi}_n^{\mathrm{CH}} \times \hat{\chi}_n^{\mathcal{I}} \times (1 - \hat{\chi}_n^{\mathrm{DB}})}{\hat{\chi}_n^{\mathrm{CH}} \times \hat{\chi}_n^{\mathcal{I}} + \hat{\chi}_n^{\mathrm{CH}} \times (1 - \hat{\chi}_n^{\mathrm{DB}}) + \hat{\chi}_n^{\mathcal{I}} \times (1 - \hat{\chi}_n^{\mathrm{DB}})} \ .$$

The term $1 - \hat{\chi}^{\mathrm{DB}}$ is used here as, despite the other cluster indices, the lowest value identifies the best cluster result for the DB index.
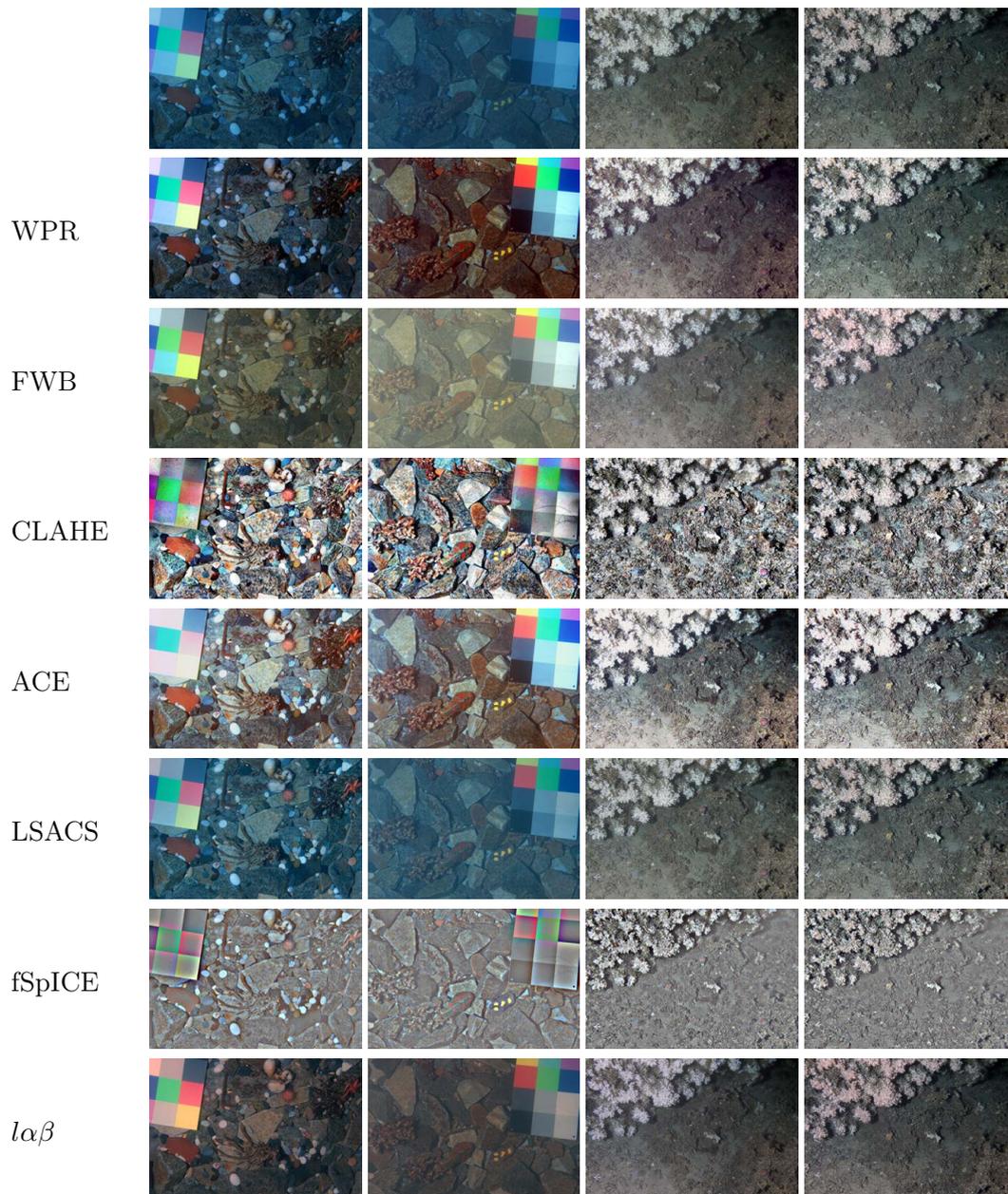
Figure 5.1: Result of applying a selection of the introduced preprocessing for four images. The first two images were recorded during a pool experiment (first and second column) and the third and forth stem from the Lofoten-Vestarelen observatory (see Section 4.2.1 for details). No individual parameter optimization was applied for the different preprocessing. Again it should be noticed that the main target of the individual algorithms is not to please the visual comfort of a human observer, but to increase the discriminability of different annotation classes and to reduce the variance within each class.

6 Computer vision for in-situ monitoring

In this thesis, computer vision (CV) is applied for environmental monitoring at the FUO LoVe. Species specific temporal and spatial information is extracted from the time-series images recorded by this FUO to assess the variation in natural behavior of different species *in-situ*. CV is applied to investigate sessile species (Section 6.1) and mobile species (Section 6.2). As an example for sessile species, the behavior of the cold water coral *Lophelia pertusa* is investigated on LoVe image set 2. Coral polyp activity (PA) is estimated and the color change of a *Lophelia pertusa* coral reef is quantified. As an example for mobile species, shrimp behavior is investigated on LoVe image set 1. Semi-transparent shrimp are identified and localized. The derived temporal and spatial distribution is visualized as pseudo color heat-maps referred to as shrimp abundance maps.

The following sections are based on previous publications. Section 6.1 is based on publication 5 and 6 and Section 6.2 is based on publication 2.

6.1 Sessile species: Lophelia pertusa

The investigated sessile species, the cold water coral *Lophelia pertusa*, is the most common reef building species in European waters [Freiwald et al., 1999]. Cold water corals can be found all around the world [Purser, 2015; Roberts et al., 2009]. The lack of a symbiotic partner allows them to grow at greater depths than tropical corals [Purser, 2015]. The *Lophelia pertusa* species was found mostly in the North Atlantic, but also throughout the Atlantic, in the Mediterranean Sea, the Gulf of Mexico, the Caribbean Sea and with little occurrences the Indian Ocean and the Pacific Ocean [Roberts et al., 2009; Zibrowius, 1980]. Records of *Lophelia pertusa* occurrences range from 39 m down to 3383 m depth [Roberts et al., 2009; Zibrowius, 1980]. The multi-spherical structure of this scleractinian species creates niches for a range of other species [Purser et al., 2013], forming a hot-spot of high local biodiversity [Purser, 2015; Roberts et al., 2009]. The preservation of *Lophelia pertusa* reefs is therefore of high interest and one cornerstone for preservation is to investigate its natural behavior.

Visual monitoring of *Lophelia pertusa* over long time periods has only been carried out to a limited extent, yet. Most previous studies have suffered from low temporal resolution, as time intervals between the inspections are either long, e.g. one year [Purser, 2015], or the temporal resolution is high but the duration of the study is limited e.g. one image every 30 minutes over 5 days and 15 hours [Buhl-Mortensen et al., 2015]. Digital images recorded by the LoVe allow for a first time to inspect the species *Lophelia pertusa* over a long time period (April 2015 - November 2015) with a temporal resolution of one image per hour (see 4.2.1 for details). Analysis of digital images by human experts is feasible if certain standards in image annotation are applied [Schoening et al., 2016b] (Chapter 3). However, the longer time period and the resulting high number of images as well as the well known problems, such as "change blindness" [Simons et al., 2005] requires an automated extraction of visual
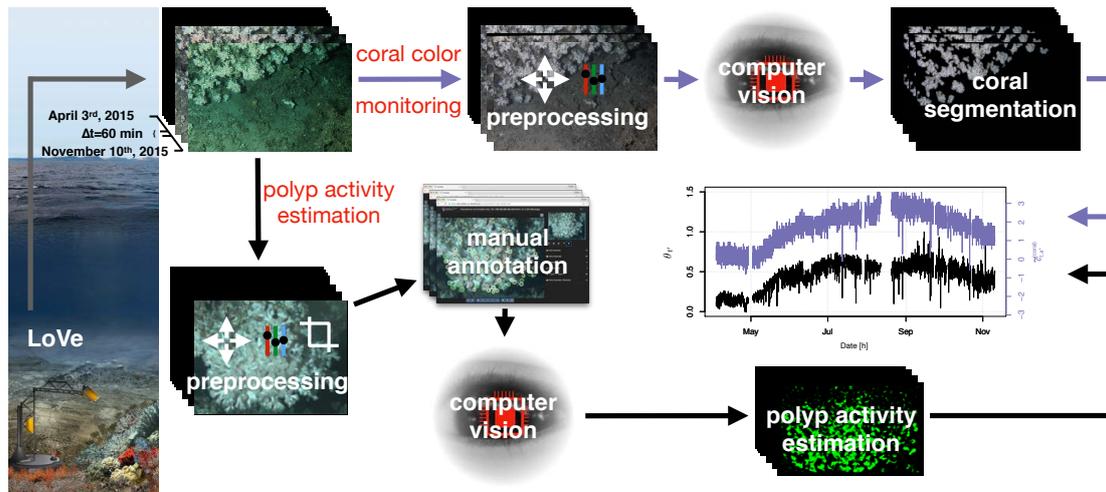
Figure 6.1: The workflow of the computational color monitoring and polyp activity (PA) estimation. Images are recorded at the LoVe observatory. The processing is then split (highlighted in red) for the two tasks of i) coral color monitoring and ii) PA estimation. The final result, a plot over time, for i) is color coded in purple and for ii) in black.

features in form of numerical values, in contrast to [Purser, 2015; Buhl-Mortensen et al., 2015] where visual features were only extracted manually.

CV is applied to automatically extract visual features relevant for the coral's state in each image. Coral features include polyp activity (task 1) and color of live corals (task 2), but also other visual features as for instance coral growth could be part of an automated visual inspection from the LoVe images. Each task requires a customized computational approach which is presented in the corresponding sections 6.1.2 and 6.1.3 and an overview of both approaches is presented in Figure 6.1.

6.1.1   Material

For the purpose of monitoring visual features of corals on the LoVe images, a time period had to be selected containing images recorded with a suitable and stable camera orientation. A suitable orientation has to feature a low variability in camera-object distances to limit the impact of the OP challenge. At the same time, it needs to cover different micro-habitats of the reef. Therefore, images from the selected period contains, areas showing live corals, large dead coral fragments and corral rubble where small coral fragments are mixed with sand (Figure 6.2). The longest time period of the second deployment with a stable camera orientation featuring the previously described requirements starts at April 3rd 2015 7:59 and ends on November 10th 2015 13:17 (LoVe image set 2). The coral monitoring is therefore carried out on images recorded in that period (for details see 4.2.1).

(a) 2015, April 3rd 7:59   (b) 2015, July 12th 1:17   (c) 2015, October 9th 7:54
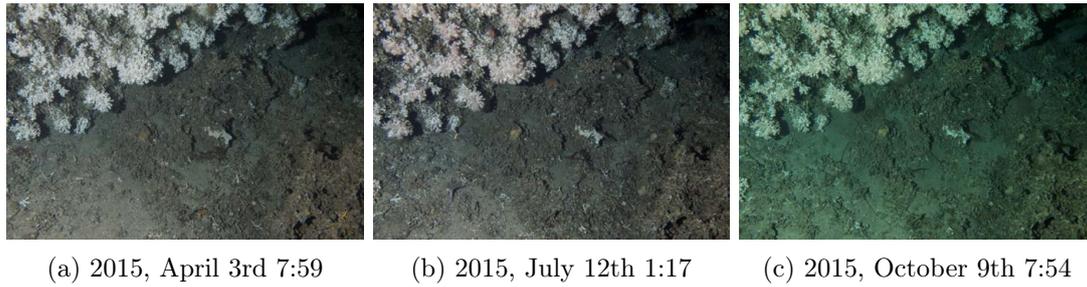
Figure 6.2: Three example images of LoVe image set 2. The camera orientation is relatively stable throughout the investigated time period. The color though is varying between the three images demanding for a color correction which increases the color constancy between different images.

6.1.2 Task 1: color monitoring

The color monitoring of the corals requires the automated extraction of color features from the digital images. The main challenges in extracting color from underwater images are caused by the OP challenge and DOP challenge. In the case of color monitoring of a sessile species using a fixed camera the effects of OP challenge are stable throughout time, as distances between camera and objects of interest are constant. Color constancy has therefore only to be achieved between different images and not within each image. Assuming that the DOP challenge is affecting the entire image equally, it can also be resolved. The image area has to cover not only the area of interest (live coral), but also other regions or objects with a different color behavior which is preferable constant. This allows to normalize the extracted color of the area of interest (Paragraph: Coral color feature extraction).

The workflow for the color monitoring includes a preprocessing of the images to optimize color constancy between images (Paragraph: Image processing for color monitoring), the automated segmentation of live coral (Paragraph: Coral segmentation) and the extraction of coral color specific numerical features (Paragraph: Coral color feature extraction) which compensate color shifts caused by changes of the in-optical properties affecting the whole image (DOP challenge).

Image processing for color monitoring   The image processing for the color monitoring $(\mathbf{I}_t \mapsto \tilde{\mathbf{I}}_t)$ consists of two steps. The digital images need to be preprocessed to achieve color constancy over time and therefore this step has to be conducted to all recorded images. A fixed white balancing (Section 5.2.2) is applied to reduce the color variance between different images $\mathbf{I}_t$. For this fixed white balancing, parameters are computed as the average of the parameters of the auto white balance (Section 5.2.1) computed on a subset of the first ten images. Computed parameters are then used for image color transformation of all $\mathbf{I}_t$. This fixed white balance allows to produce "normal" colors now with an increased color constancy over all $\mathbf{I}_t$. Furthermore, images are spatially aligned to compensate camera movement using the enhanced correlation coefficient (ECC) maximization method [Evangelidis et al., 2008].

A mean image is computed from the first ten images $\{\tilde{\mathbf{I}}_1, ..., \tilde{\mathbf{I}}_{10}\}$ which are aligned to $\tilde{\mathbf{I}}_1$:

$$\bar{\mathbf{I}}(x, y, c) = \frac{1}{10} \sum_{t=1}^{10} \tilde{\mathbf{I}}_t(x, y, c)$$

All images are then aligned to $\bar{\mathbf{I}}$. The image alignment allows to enhance the later segmentation by computing a binary image, masking only those pixels which where segmented as coral in all images as "coral pixels".

Coral segmentation The coral segmentation is performed by applying the unsupervised ML algorithm, the hierarchical hyperbolic self organizing map (H^2SOM) algorithm (Section 2.7.1: Hierarchical Hyperbolic Self Organizing Map), which also has been used the calcareous algae segmentation (Chapter 7.1), for cold water coral segmentation in ROV video frames [Purser et al., 2009] and for poly-metallic nodule segmentation [Schoening et al., 2015; Schoening et al., 2016a]. It is applied similar as described in [Purser et al., 2009]. Wavelet responses using 6 orientations and 3 scales (Section 2.3.4) are computed for each image forming the 18 dimensional feature space. The features are then clustered using the H^2SOM algorithm with 3 rings, 8 neighbors, an exponential learning rate and an Euclidean metric. Using a fixed camera for monitoring a sessile species also allows to simplify the object detection. A conservative approach, which prefers precision (i.e. avoiding false positives (FP)) over recall (i.e. avoiding false negatives (FN)), can be applied. Hence, the selection of H^2SOM prototypes representing coral in the feature space, is performed as described in Section 7.1.2. This selection is additionally manually optimized on a small subset of about ten images distributed randomly over the entire time period. After this H^2SOM training and prototype selection, for each image $\tilde{\mathbf{I}}_t$ image pixels $\tilde{\mathbf{I}}_{t,(x,y)}$ are classified into "coral pixels" ($= 1$) and "non-coral pixels" ($= 0$). The resulting binary images $M_1^{(\text{coral})}, ..., M_t^{(\text{coral})}$ with $\mathbf{M}_t^{(\text{coral})} \in \{0,1\}^{w \times h}$ ($w = 5184, h = 3456$) are used to compute the static coral area mask $\hat{\mathbf{M}}^{(\text{coral})}$ as the intersect of all image masks $\mathbf{M}_t^{(\text{coral})}$ by a pixel-wise multiplication of all binary classification masks, where the elements of $\hat{\mathbf{M}}^{(\text{coral})}$ are defined as:

$$\hat{\mathbf{M}}^{(\text{coral})}(x, y) = \prod_t \mathbf{M}_t^{(\text{coral})}(x, y). \tag{6.1}$$

Coral color feature extraction The segmentation mask $\hat{\mathbf{M}}^{(\text{coral})}$ is used to extract color features of two different areas and the full image for every image $\tilde{\mathbf{I}}_t$. The first area covering live coral and the second the not live part in the images. Color features are extracted after transforming the RGB images to the CIELab color space: $\tilde{\mathbf{I}}_t \mapsto \tilde{\mathbf{I}}_t^{(L,a,b)}$ (Section 2.2). Averages are computed from every transformed image $\tilde{\mathbf{I}}_t^{(L,a,b)}$ for the full image:

$$\bar{\mathbf{c}}_t^{\text{full}} = \frac{1}{w \cdot h} \sum_{x=1}^{w} \sum_{y=1}^{h} \tilde{\mathbf{I}}_t^{(L,a,b)}(x, y)$$

for the live coral areas:

$$\bar{\mathbf{c}}_t^{(\mathrm{coral})} = \frac{1}{w \cdot h} \sum_{x=1}^{w} \sum_{y=1}^{h} \hat{\mathbf{M}}^{(\mathrm{coral})}(x,y) \cdot \tilde{\mathbf{I}}_t^{(L,a,b)}(x,y)$$

and for the non-coral area:

$$\bar{\mathbf{c}}_t^{(\neg\mathrm{coral})} = \frac{1}{w \cdot h} \sum_{x=1}^{w} \sum_{y=1}^{h} (1 - \hat{\mathbf{M}}^{(\mathrm{coral})}(x,y)) \cdot \tilde{\mathbf{I}}_t^{(L,a,b)}(x,y).$$

Due to the absence of a color reference in the images, color normalization without external information is needed. Color shifts affecting the whole image, caused by changes of the in-optical properties (DOP challenge), are resolved by color normalization: $\bar{\mathbf{c}}_t^{(\mathrm{coral})}$ and $\bar{\mathbf{c}}_t^{(\neg\mathrm{coral})}$ are normalized by subtracting the mean color of the full image $\bar{\mathbf{c}}_t^{\mathrm{full}}$:

$$\widetilde{\mathbf{c}}_t^{(\mathrm{coral})} = \bar{\mathbf{c}}_t^{(\mathrm{coral})} - \bar{\mathbf{c}}_t^{(\mathrm{full})} \tag{6.2}$$

and

$$\widetilde{\mathbf{c}}_t^{(\neg\mathrm{coral})} = \bar{\mathbf{c}}_t^{(\neg\mathrm{coral})} - \bar{\mathbf{c}}_t^{(\mathrm{full})} \tag{6.3}$$

6.1.3 Task 2: polyp activity monitoring

The monitoring of the Polyp Activity (PA) of corals requires the automated estimation of the PA in each recorded image. In contrast to the color monitoring, the OP challenge and DOP challenge are not the main challenges, as PA estimation is not based on color only. Nevertheless, the impact of OP challenge is reduced by a preprocessing (Image preprocessing) and by focusing on a region of the images with live corals located towards the center of the image. This region is well illuminated, features lowest chromatic aberration, lens distortion and variance of camera-object distance (Chapter 4). The main challenge in estimating the PA is the limited resolution of the images. It is very close to the limit allowing to classify the activity level of individual polyps manually. Hence, the annotation challenges AQ, AC and AU have to be solved.

The customized computational approach requires more manual interaction than the color monitoring approach. The workflow for the PA estimation consists of three main steps. First, a gold standard is constructed for different level of PA. Hence, a manual annotation study is carried out by three observers on a region of interest in 13 images (Paragraph: Manual annotation). In the second step, a convolutional neural network (CNN) is trained using the gold standard to automate the PA classification (Paragraph: Deep learning). Finally in the third step, the trained network is integrated into an algorithmic estimation of PA in the region of interest on all images (Paragraph: Polyp activity estimation).

Image preprocessing Images need to be preprocessed to increase color constancy over time and therefore for all recorded images to reduce the variation of appearance within the different polyps state (inner class variance). The image processing for the PA estimation $(\mathbf{I}_t \mapsto \tilde{\mathbf{I}}'_t)$ is similar to the image preprocessing for the coral color monitoring. Preprocessing only differs in the parameters of the fixed white balance, which was chosen manually for the PA monitoring. Small camera movement caused by changing currents are again compensated by applying the ECC algorithm. Image registration is necessary to allow a comparison of PA states on a pixel level in consecutive images.

Manual annotation Placing annotations in underwater images manually is a challenging process (Chapter 3). Manual identification of PA appeared to be an even more challenging and time consuming task for the LoVe images. One reason for this is the relatively low pixel resolution of the polyps (not bigger than 45×45 pixels) in the images. Furthermore, distinguishing between different level of activity is only possible by comparing the same polyps at different time points. The manual annotations of the coral polyps were carried out using BIIGLE 2.0 (Section 3.3). The software allows the user to flip quickly between images making the identification of the different gradients of PA possible. The human observers can learn the different characteristics for different polyps of different levels of PA.

The manual annotation here is a whole image annotation (WIS) task (Section 3.1). Similar as described in Section 3.1, a manual single polyp annotation $a_t^{(i)}$ consists of a region $\mathbf{r}_t^{(i)}$ in an image $\mathbf{I}_t$, a class label $\omega_t^{(i)}$, and an observer o.

$$a_t^{(i)} = \left(\mathbf{r}_t^{(i)}, \omega_t^{(i)}, o \right)$$

The latter is the person who created the annotation. Here, a region $\mathbf{r}_t^{(i)}$ is a subset of pixels from an image $\tilde{\mathbf{I}}'_t$, has a circular shape and is described by a center point $(x_t^{(i)}, y_t^{(i)})$ and a radius $r_t^{(i)}$:

$$\mathbf{r}_t^{(i)} = \left(x_t^{(i)}, y_t^{(i)}, r_t^{(i)} \right)$$

For this study, three biologist experts (JJ, IN and PBM), provided the necessary manual annotations. A consensus expert method could therefore be applied here (Section 3.2.1). A region of interest in the images was selected for the manual annotation. This region is located in the middle of the coral area and shows mostly live corals (Figure 6.3). In a preliminary study, conducted to examine the ability of identifying different level of PA manually in the LoVe images, this region showed the highest number of manual annotations. All biologist experts, i.e. observers, were able to identify the activity levels for most of the polyps in this region. Furthermore, the region is one of the live coral areas with the highest sharpness, the lowest chromatic aberration, lowest lens-distortion and lowest variation of camera-object distance in all images. The impact of the OP challenge is limited in this region. Hence, the automated PA estimation is also limited to this region of interest using a classification mask (see below).
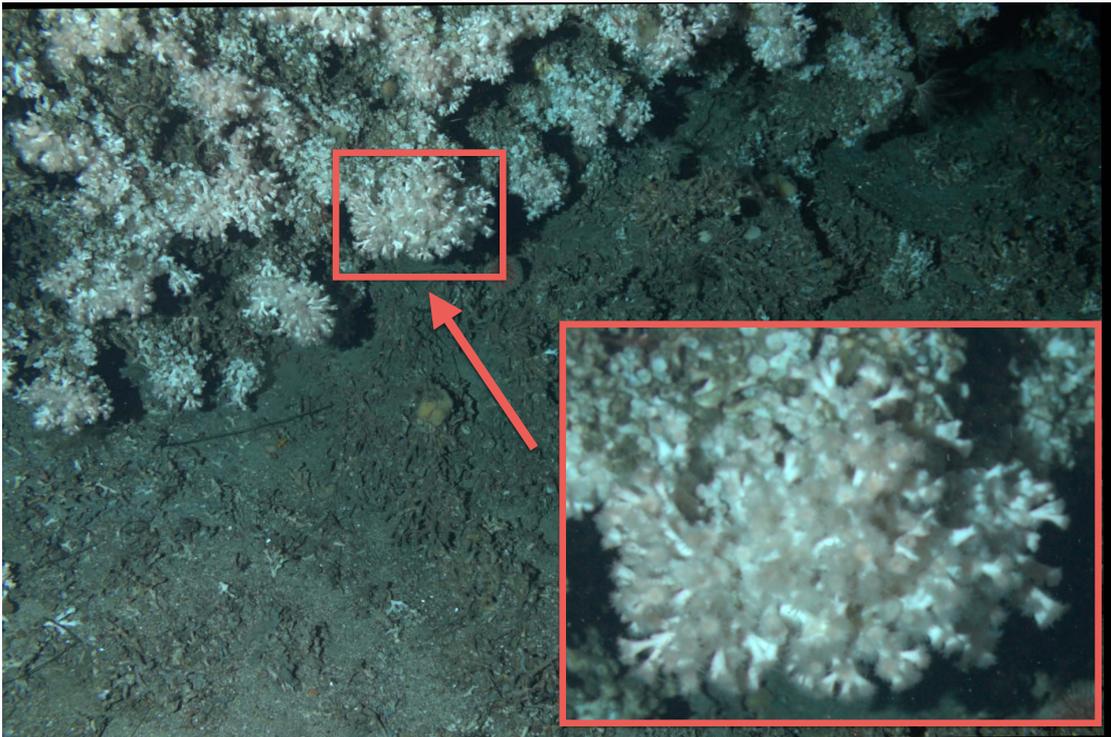
Figure 6.3: The selected region of interest for the manual annotation of PA and it location in an example image. The automated estimation of the pixel wise PA is also limited to this region which is highlighted in red. The displayed image is color corrected and aligned as described in Section 6.1.3 (Image preprocessing).

The manual annotation process is carried out in two steps: first, an initial annotation process and second, a re-annotation process after a break of one week. As described in Chapter 3 this two step approach allows to evaluate the quality of the manual annotations of single observers. In each step, observers are allowed to place circular annotations by setting a position (x, y) and radius r in a subset of images $\{\tilde{\mathbf{I}}'_{t'}\}_{t'=1,\dots,13}$, i.e. the annotation images, using BIIGLE 2.0 (Section 3.3). t' instead of t is used as an index here indicating that the subset images do not have to be consecutive images. Experts also have to select a class ω for each annotation. ω can be chosen from a list of six classes: $\omega \in \{$"Fully Expanded", "Fully / Partly Expanded", "Partly Expanded", "Partly Expanded / Retracted", "Retracted", or "Recently Dead"$\}$, where the first five classes can be interpreted as a gradient from maximum to minimum activity. Annotation classes were selected and defined in a preliminary study. In the initial annotation process each observer is independently annotating 13 images from the actual time period. 10 images were chosen randomly and 3 were selected manually showing low PA to increase the number of annotated polyps with corresponding ω. During this process, each observer can only see their own annotations. After a break of one week (experts were not allowed to view the images during the break) a subset of 3 of the 13 initial images are re-annotated by each expert individually .

Gold standard   The annotations from the first annotation step are used to create a gold standard $G = \{g_t^{(j)}\}_{j=1,\dots,|G|}$ as described in Section 3.2.1. Similar to annotations $a_t^{(i)}$, each $g_t^{(j)}$ consists again of a circular polyp region $\bar{\mathbf{r}}_t^{(j)}$ in the corresponding image $\tilde{\mathbf{I}}'_t$ and a class label $\omega_t^{(j)}$:

$$g_t^{(j)} = \left(\bar{\mathbf{r}}_t^{(j)}, \omega_t^{(j)} \right)$$

Different observers' annotations from the same image $\tilde{\mathbf{I}}'_{t'}$ are compared. A set $\Lambda_t^{(j)}$ of overlapping annotations $a_t^{(i)}, a_t^{(i')}$ satisfying

$$||(x_t^{(i)}, y_t^{(i)}) - (x_t^{(i')}, y_t^{(i')})|| \leq \min \left(\frac{r_t^{(i)}}{2}, \frac{r_t^{(i')}}{2} \right)$$

and $\omega^{(i)} = \omega^{(i')}$ from at least two different observers $(2 \leq |\Lambda^{(j)}| \leq 3)$ form the basis for a gold standard annotation $g_t^{(j)}$. In other words, the regions $\mathbf{r}_t^{(i)}, \mathbf{r}_t^{(i')}$ from $a_t^{(i)}, a_t^{(i')}$ have an overlap of at least $\frac{2}{3}$. In order to increase the reproducibility and the number of annotations per class, classes are merged to "active" $(\hat{\omega} = 1)$ and "retracted" $(\hat{\omega} = 0)$ in the gold standard. The class "active" merges "Fully Expanded", "Fully / Partly Expanded" and "Partly Expanded", the class "retracted" merges "Retracted" and "Recently Dead". The class "Partly Expanded / Retracted" is discarded from the gold standard. Each gold standard region $\bar{\mathbf{r}}_t^{(j)}$ is computed
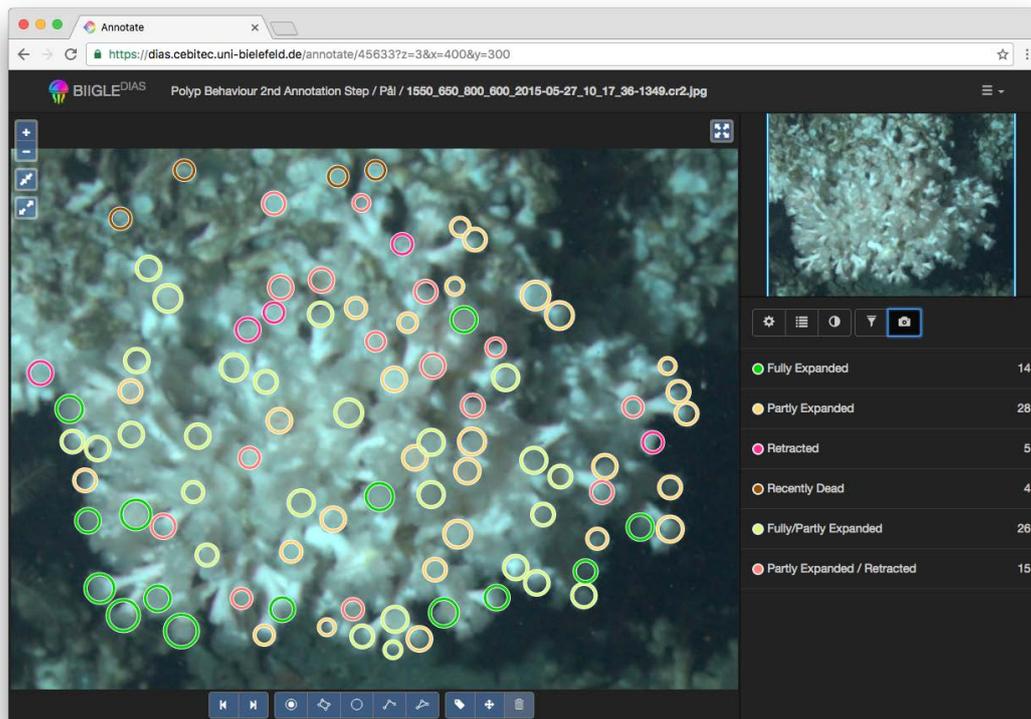
Figure 6.4: A screen-shot from the annotation software BIIGLE 2.0. The software was used to generate the manual annotations of the different level of PA. The displayed image is one of the 13 images used in the manual annotation process. Colors appear to be a little more greenish, due to the manually selected parameters for the fixed white balance. Annotation classes are presented on the lower right (color, name, number of annotations with referring class label). Most polyps are annotated as "Partly Expanded" or "Fully / Partly Expanded".

Figure 6.5: The process of creating a classification mask for the PA estimation. All computed gold standard annotations $\{g^{(j)}\}$ are visualized on the left. Colors of $g^{(j)}$ indicate the annotation class (Figure 6.4 for color coding). The set of gold standard annotations G is generated from the manual annotations of the first annotation step (see text for details). The binary image in the center is computed from the $\{\bar{\mathbf{r}}^{(j)}\}$, the regions of the gold standard annotations. The binary image is modified using the morphological closing operator $\phi^{(\text{closing})}$, followed by a manual closing of the remaining holes in the mask (on the right). This mask is used to limit the target area for the automated PA estimation.

as:

$$
\begin{aligned}
\bar{\mathbf{r}}_t^{(j)} &= \left(\bar{x}_t^{(j)}, \bar{y}_t^{(j)}, \bar{r}_t^{(j)} \right) \text{ with} \\
\bar{x}_t^{(j)} &= \frac{1}{|\Lambda_t^{(j)}|} \sum_{a_t^{(i)} \in \Lambda_t^{(j)}} x_t^{(i)}, \\
\bar{y}_t^{(j)} &= \frac{1}{|\Lambda_t^{(j)}|} \sum_{a_t^{(i)} \in \Lambda_t^{(j)}} y_t^{(i)} \text{ and} \\
\bar{r}_t^{(j)} &= \frac{1}{|\Lambda_t^{(j)}|} \sum_{a_t^{(i)} \in \Lambda_t^{(j)}} r_t^{(i)}.
\end{aligned}
$$

Furthermore, the computed polyp regions $\bar{\mathbf{r}}$ provide the basis to compute a mask $\mathbf{M}^{\text{polyps}}$, masking all polyp positions in the images. $\mathbf{M}^{\text{polyps}}$ is used to limit the target area for the automated estimation of PA (Figure 6.5).

Deep learning The gold standard annotations G provide the basis for the training of the ML algorithm, in this case a deep learning algorithm (Section 2.7.3). In order to apply deep learning, for each $g^{(j)}$ one image patch $\rho^{(j)}$ is extracted at the center of each $\bar{\mathbf{r}}^{(j)}$. All image patches have the same size ($|\rho^{(j)}| = 46 \times 46$ pixel), which is defined by the largest $\bar{\mathbf{r}}^{(j)}$ (i.e. largest diameter). As no "background" class exists from the manual annotation process, a background class, not showing polyps, is created by randomly extracting image patches from all $\{\tilde{\mathbf{I}}'_{t'}\}_{t'=1,\dots,13}$. Background patches have the same size as the patches extracted at $\{\bar{\mathbf{r}}^{(j)}\}$. Each background location is set to have a minimal distance of 20 pixels to all $\bar{\mathbf{r}}^{(j)}$. For each of the 13 annotation images 100 background patches are generated. The number of examples for each ML class is further increased by data augmentation (Section 2.6). Extracted patches

are flipped, rotated and Gaussian noise is added to them, resulting in a data augmentation factor of 48 (12 rotations). A larger number of examples usually results in models which generalize better, specifically for deep learning [He et al., 2016; Witten et al., 2016].

In principle the CNN is trained with three different classes, which are "active", "retracted" and "background", but for the sake of compactness it is not differentiated between the last two classes "retracted" and "background". It is sufficient enough to differentiate only between $\hat{\omega} = 1$ ("active") and $\hat{\omega} = 0$ ("not active"). For each ML class $\hat{\omega} \in \{0, 1\}$ the finally obtained boosted image patches are divided into training (70%), validation (20%) and test (10%) patches. Validation patches are used to optimize the training parameters, while the test patches are only used for the final evaluation.

A deep convolutional neural network (CNN) with a LeNet-5 [LeCun et al., 1998] layout (Section 2.7.3 LeNet-5) is trained on the training patches using the NVIDIA DIGITS software (https://developer.nvidia.com/digits).

Polyp activity estimation   PA estimation is carried out using the C++ interface of the Caffe framework [Jia et al., 2014]. The trained classifier is used to classify each pixel $\tilde{\mathbf{I}}'_t(x, y)$ that has been marked positive in $\mathbf{M}^{\text{polyps}}$ ($\mathbf{M}^{\text{polyps}}(x, y) > 0$). To this end, a patch $\hat{\rho}_t^{(x,y)}$ of size 46×46 pixel) centered at (x, y) is shown to the network and the result ($\Delta(\rho_t^{(x,y)}) = 1$ for "active" and $\Delta(\rho_t^{(x,y)}) = 0$ for "retracted" or "background") represents the PA position (x, y). The aggregated number of patches classified to be "active" in relation to the size of the classification area $\sum_{x,y} \mathbf{M}^{\text{polyps}}(x, y)$ gives the estimated PA θ_t for each image $\tilde{\mathbf{I}}'_t$:

$$\theta_t = \frac{1}{\sum_{x,y} \mathbf{M}^{\text{polyps}}(x, y)} \sum_{x,y} \Delta(\rho_t^{(x,y)}). \tag{6.4}$$

Evaluation   In order to evaluate the manual annotations for reproducibility, the inter- and intra-observer agreement is computed (Section 3.2.3). The annotation task for the observers was to annotate polyps only if they could identify the activity-level. As a result, the amount of annotations per observer varies and the inter- and intra-observer agreement is only computed for overlapping annotations. These scores will not reflect the ability to place accurate and reproducible annotations on polyps in general. It only reflects the quality of a derived gold standard (see above).

Evaluating the accuracy of the estimated PA θ_t is a non-trivial task. To understand the performance of the trained network and to allow to interpret its output, the evaluation contains three steps: first, the evaluation on the generated image patches $\rho^{(j')}$, second the evaluation on a pixel level for each $\bar{\mathbf{r}}^{(j)}$ and third the evaluation of the finally estimated PA θ_t for each image $\tilde{\mathbf{I}}'_t$.

The basic accuracy measurement for a CV algorithm can be carried out on the data, used for training, validation, and testing. As stated in Section 2.8 the validation set is here used to optimize training parameters and the test set is only used for the final evaluation.

The accuracy of the network on this data, i.e. image patches with a annotation class $\hat{\omega}$, is computed as described in Section 2.10 using the accuracy (ACC) (eq. 2.2) with:

$$\text{TP} = |\Delta(\rho^{(j')}) = 1 \wedge \hat{\omega}^{(j')} = 1|,$$

$$\text{TN} = |\Delta(\rho^{(j')}) = 0 \wedge \hat{\omega}^{(j')} = 0|,$$

$$\text{FP} = |\Delta(\rho^{(j')}) = 1 \wedge \hat{\omega}^{(j')} = 0|,$$

$$\text{FN} = |\Delta(\rho^{(j')}) = 0 \wedge \hat{\omega}^{(j')} = 1|$$

Evaluating the accuracy and significance on a pixel level is more complicated. The ground truth data need to be generated for the images on a pixel level. Here, it is generated using the gold standard annotations G. The performance of the network on a pixel level for each $g^{(j)}$ is estimated with a recall and false positive rate (FPR). These performance scores are applied as no information of the level of activity is available for other areas. For each $\bar{\mathbf{r}}^{(j)}$ the relative number of the classification outputs $\Delta(\rho_t^{(x,y)})$ on pixel level is computed:

$$\kappa(g^{(j)}) = \frac{\sum_{(x,y) \in \bar{\mathbf{r}}^{(j)}} \Delta(\rho_t^{(x,y)})}{|\bar{\mathbf{r}}^{(j)}|}. \tag{6.5}$$

A recall estimate is then computed as:

$$\text{recall} = \sum_{g^{(j)} : \hat{\omega}^{(j)} = 1} \kappa(g^{(j)}) \tag{6.6}$$

and the FPR is estimated as:

$$\text{FPR} = \sum_{g^{(j)} : \hat{\omega}^{(j)} = 0} \kappa(g^{(j)}). \tag{6.7}$$

Finally, in order to evaluate the performance on the entire masked image, gold standard annotations and classification output are compared for each annotated image $\tilde{\mathbf{I}}'_{t'}$ with $t' = 1, ..., 13$. The relative number of pixels enclosed in the gold standard with $\omega = $ "active" per image $\tilde{I}'_{t'}$

$$\varphi_{t'} = \frac{1}{\sum_{x,y} \mathbf{M}^{\text{polyps}}(x,y)} \sum_{j \in A_{t'}} |\bar{\mathbf{r}}^{(j)}| \tag{6.8}$$

with $A_{t'} = \{j | \omega^{(j)} = $ "active"$, \bar{\mathbf{r}}^{(j)} \subset \tilde{\mathbf{I}}'_{t'}\}$ is compared with the relative number of pixels classified as "active" $\theta_{t'}$ (Equation 6.4).

6.1.4 Results

The results of the color monitoring workflow (task 1) and PA monitoring workflow (task 2) are presented separately.
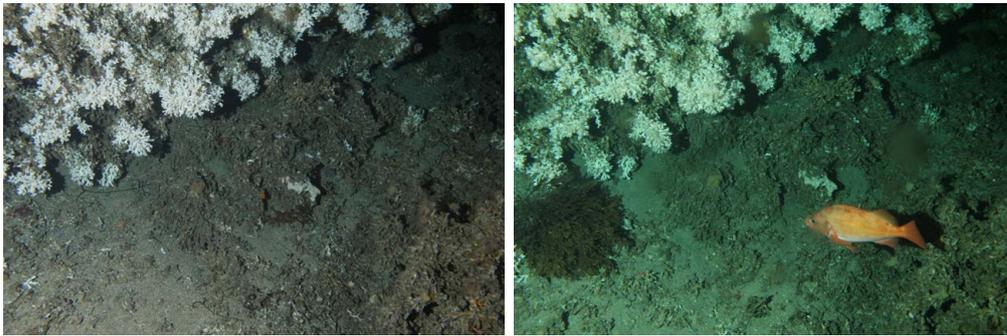
(a) Original images
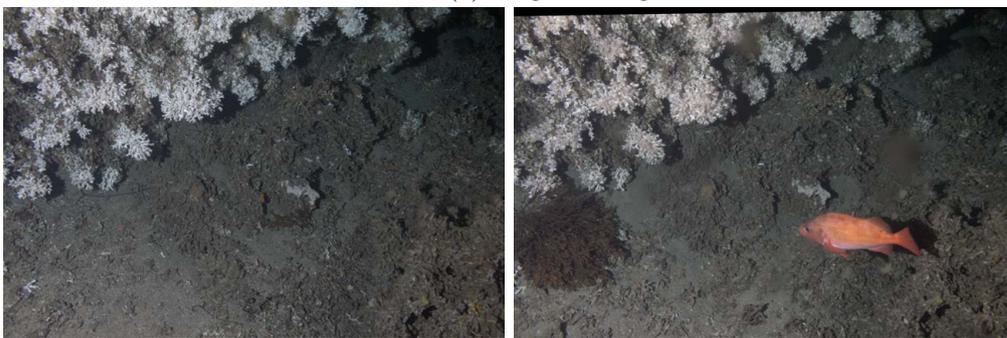
(b) Preprocessed images $(\mathbf{I} \mapsto \tilde{\mathbf{I}})$

(c) Preprocessed images $(\mathbf{I} \mapsto \tilde{\mathbf{I}}')$

Figure 6.6: Two examples illustrating the need for image preprocessing. The upper row shows the original images (a), the second row shows the corresponding preprocessed images $(\mathbf{I} \mapsto \tilde{\mathbf{I}})$ as described in 6.1.2 (b) and the lower row shows the corresponding preprocessed images $(\mathbf{I} \mapsto \tilde{\mathbf{I}}')$ as described in 6.1.3. The processing of images (b) and (c) includes the applying of a fixed white balance and spatial alignment. The only difference between (b) and (c) is the use of different parameters for the fixed white balance, resulting different color shifts for the different preprocessing. The originals in (a) show that the color of the objects present in the images can have an influence on the color of the whole image. The lower images (b) and (c) show that the use of a fixed white balance resolves this challenge.

Figure 6.7: The coral feature extraction mask $\hat{\mathbf{M}}^{\text{coral}}$. The mask is computed by multiplying individual image masks $\hat{\mathbf{M}}_t^{\text{coral}}$ computed using unsupervised ML as described in Section 6.1.2.

Task 1: Color monitoring   The image processing for the color monitoring $(\mathbf{I} \mapsto \tilde{\mathbf{I}})$ includes the processing with a fixed white balance, which improved the color constancy throughout all images. It reduced the effects in terms of color change caused by e.g. fish appearance (Figure 6.6b). The included alignment produced reliable results for the analyzed time period of more than half a year.

Next, coral was segmented in each $\tilde{\mathbf{I}}_t$ using the coral mask $\hat{\mathbf{M}}^{(\text{coral})}$ (Figure 6.7, Equation 6.1). The mask does not show any false positive pixels and represents a rather conservative segmentation.

Coral color features were extracted for all of the $N = 4862$ images. In Figure 6.8 the a* channels are presented for the live coral $(\widetilde{c}_{t,a*}^{(\text{coral})})$, non-coral area $(\widetilde{c}_{t,a*}^{(\neg\text{coral})})$ and the full image $(\bar{c}_{t,a*}^{(\neg\text{full})})$. Looking at the time lines $\widetilde{c}_{t,a*}^{(\text{coral})}$, $\widetilde{c}_{t,a*}^{(\neg\text{coral})}$ and $\bar{c}_{t,a*}^{(\neg\text{full})}$, only $\widetilde{c}_{t,a*}^{(\text{coral})}$ shows a long-term change of color over time. The $\widetilde{c}_{t,a*}^{(\text{coral})}$ is increasing from April until mid-August before it decreases again until the end of the analyzed time period. This long-term change can be interpreted as a change of redness of the coral, as a higher $a*$-channel results in a color shift towards red. The selected feature $\widetilde{c}_{t,a*}^{(\text{coral})}$ can therefore be used to monitor change of redness of the *Lophelia pertusa* reef over time.

Other short-term events can be identified in the time series, noticeable as outliers. One event, event A in Figure 6.8, occurred during late April and could be identified as a change of the inherent optical properties of the water possibly caused by algae-bloom. Other single image outliers could be identified as a group of fishes, event B in Figure 6.8 (July 23rd

09:17:36), and a short-lasting (less than 2 hours) high water turbidity, event C in Figure 6.8 (November 3rd 01:17:22). The measured colors of the coral rubble (Equation 6.3) and live coral area (Equation 6.2) are not affected by these occasional events (A, B and C), as intended. The straightforward normalization of the extracted colors allows to focus on the long-term changes.

Task 2: polyp activity monitoring The image preprocessing for PA monitoring $(\mathbf{I} \mapsto \tilde{\mathbf{I}}')$ is similar to the preprocessing for color monitoring. The use of a manually chosen fixed white balance also improved the color constancy throughout all images. Compared to $\mathbf{I} \mapsto \tilde{\mathbf{I}}$ images show a small color shift towards blue (Figure 6.6c). The image alignment reduced the effect of camera shifting (spatial alignment), as done for in the image processing for color monitoring.

Next, the activity of polyps were annotated manually in $\tilde{\mathbf{I}}'_{t'}$. Resulted manually annotations were evaluated for inter- and intra observer agreement to evaluate the reproducibility of the manual annotations (Section 3.2.3). The computed inter-observer agreement reached an average value of 0.64 and the intra-observer agreement an average of 0.69 over all annotation classes. Reducing the 6 classes to two ("active" and "retracted") resulted in an average inter-observer agreement of 0.86 and an average intra-observer agreement of 0.89. Manual annotations were used to create a gold standard for the training of a ML algorithm. A total 4910 annotations were placed by the observers of which 1957 could not be used as the showed no overlap, low overlap or class differences. The reaming annotation were combined to create $|G| = 1110$ gold standard annotations, including 252 $g^{(j)}$ with $\omega =$ "retracted".

Next, all $g^{(j)}$ were used to create image patches for the training and testing of a deep learning algorithm, a CNN. The trained network was evaluated on the training-, validation- and test-patches. An accuracy (Equation 2.2) of 0.98 was computed for the training patches. The accuracy computed for validation- and test-patches patches reached a score of 0.96 each.

As described in Section 6.1.3, network performance was additionally evaluated for each gold standard annotation $g^{(j)}$. The number of pixels classified to be active is higher for $g^{(j)}$ with the $\omega^{(j)} =$ "active" than for those with $\omega^{(j)} =$ "retracted" (Figure 6.9b). An average recall (Equation 6.6) of 0.54 and an average FPR (Equation 6.7) of 0.09 was computed. Examples of original image patches and resulted pixel classification for "active" and "retracted" are displayed in Figure 6.9a.

The classification performance on the whole images $\{\tilde{\mathbf{I}}'_{t'}\}$ was evaluated by comparing $\varphi_{t'}$ (Equation 6.8) with $\theta_{t'}$ (Equation 6.4) (Figure 6.9c). The amount of pixels classified as "active" is lower than the amount of pixels in the gold standard annotations with $\omega =$ "active" for images with a general low PA, but higher for images with high PA. A Spearman's rank correlation coefficient of 0.95 was estimated for $\varphi_{t'}$ and $\theta_{t'}$.

The network was finally used to estimate the PA for all available images and the estimated relative pixel activity is presented in Figure 6.10 for the whole time period (more than half a year). Looking at the whole time period, the lowest activity can be noticed during an algae
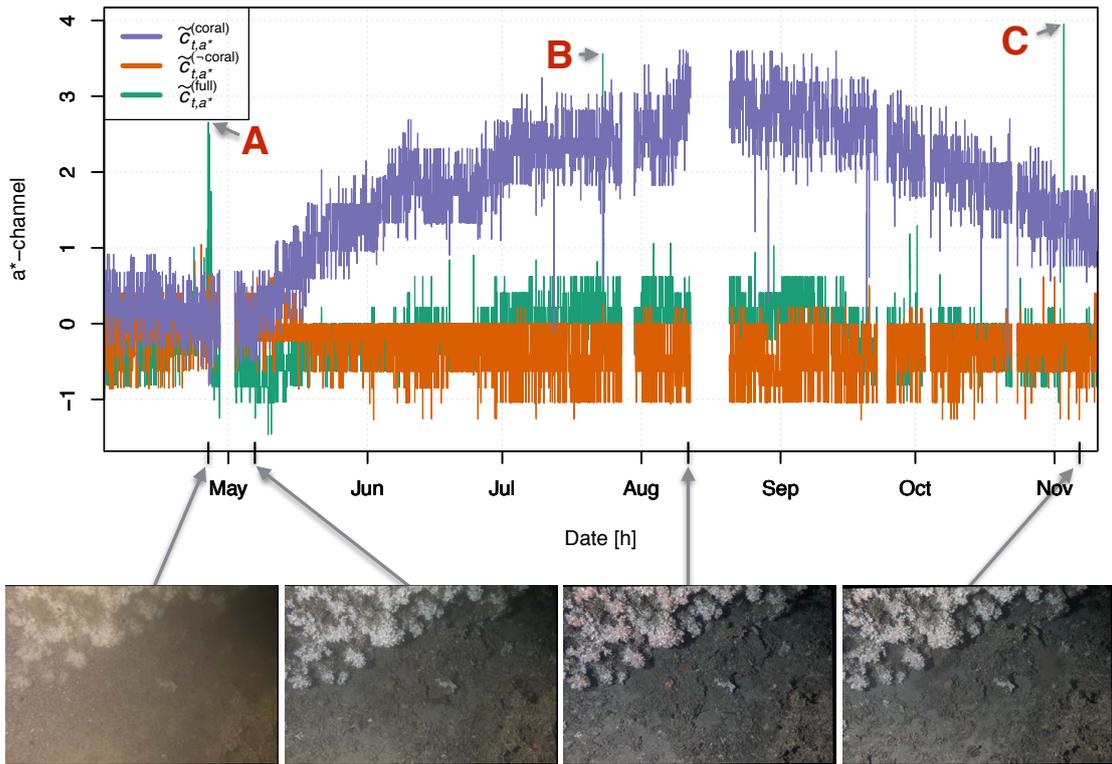
Figure 6.8: The time series of $\widetilde{c}_{t,a*}^{(\text{coral})}$, $\widetilde{c}_{t,a*}^{(\neg\text{coral})}$ and $\bar{c}_{t,a*}^{(\neg\text{full})}$. Features extracted for the live coral area and the non-coral area were normalized by subtracting the corresponding mean color of the full image (Equation 6.2 and 6.3). Several short-term events can be identified in $\bar{c}_{t,a*}^{(\neg\text{full})}$ highlighted with letters A, B and C. The A event in late April is affecting the whole image, as the optical properties of the water were changing (see first two example images below the graph). Events B and C are fish occurring in the camera frame. $\widetilde{c}_{t,a*}^{(\text{coral})}$ is the only time series showing a long-term change, the color of the live coral area is changing over time. Live corals appear to be more reddish in August than in the beginning of April and again less reddish in November.

(a)

(b)

(c)
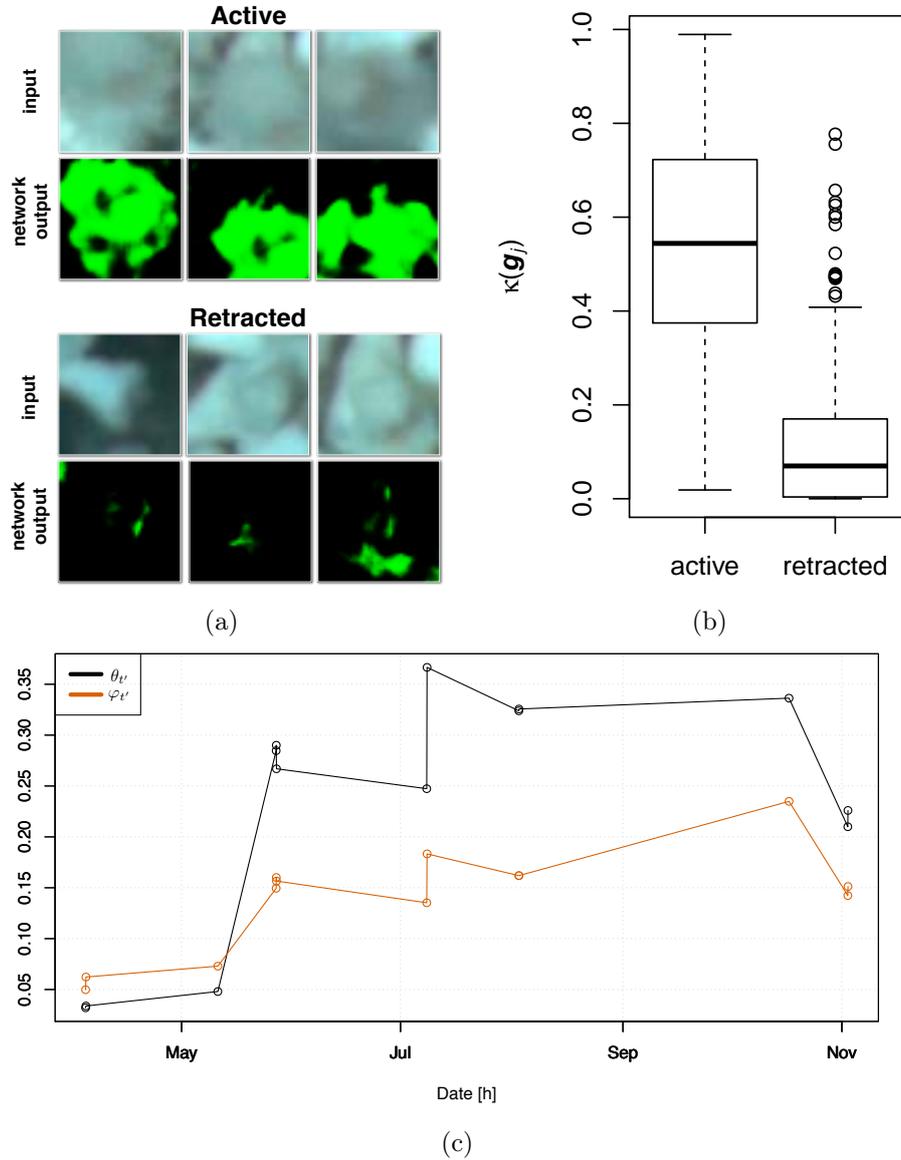
Figure 6.9: The analysis of the network performance for the PA classification. In (a) examples of three active and three "retracted" patches and below their corresponding classification outputs on a pixel level are displayed. Patch positions classified as 1 ($\Delta(\rho_t^{(x,y)} = 1$) are marked in green. In (b) boxplots are presented showing the $\kappa(g^{(j)})$ (Equation 6.5) values grouped for $\omega^{(j)} =$ "active" and $\omega^{(j)} =$ "retracted" (Equation 6.5). The Median, the upper and the lower quartile differ significantly between the two ML classes. In (c) the relative amount of pixels of all $g^{(j)}$ with $\omega^{(j)} =$ "active" $\varphi_{t'}$ (Equation 6.8) (colored circles) is compared with the relative amount of pixels classified as "active" $\theta_{t'}$ (Equation 6.4) (black circles) for each of the 13 images form the manual annotation study. In order to visualize the time trend and to identify the temporal order of the images estimated values are connected by linear interpolated lines. $\theta_{t'}$ is lower than $\varphi_{t'}$ for images with low activity, but higher for images with high activity. A Spearman's rank correlation coefficient of 0.95 was estimated for $\varphi_{t'}$ and $\theta_{t'}$.
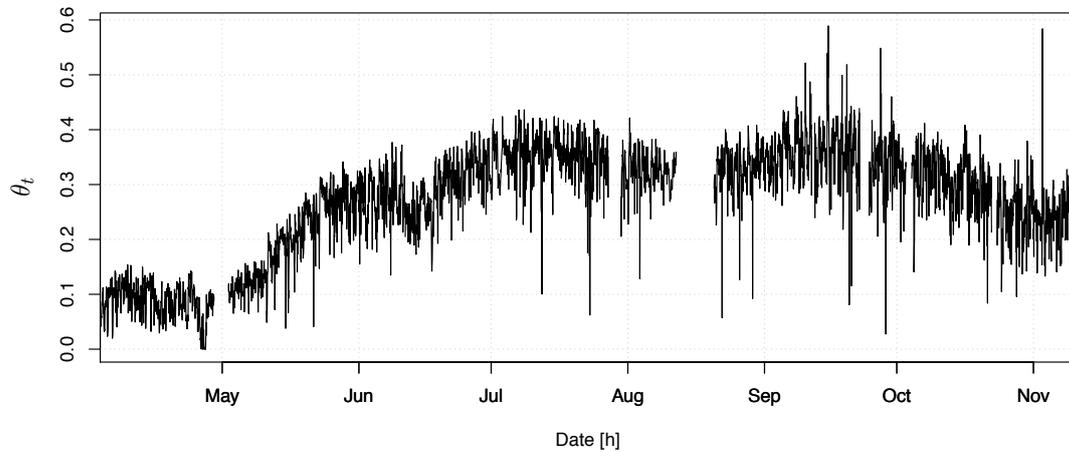
Figure 6.10: Plot for the estimated PA θ_t of the full time scale (early April 2015 until mid-November 2015). θ_t was estimated pixel wise on each sub-image $\tilde{\mathbf{I}}'_t$ showing the relative number of pixels classified as active. θ_t can be interpreted as the relative amount of active polyps. Gaps in the time series stem from camera malfunctions. The two main gaps in image recording occurred in end-April and in mid-August. θ_t is low in the beginning of the time scale. It is increasing from late April until mid-Mai. Activity stays on a high level and tends to decrease again in the beginning of October.

bloom event at the end of April. The amount of particles in the water is so high that the activity estimation fails. After this event the activity increases, having a first peak in the first half of June. The next peak is reached around middle of July. After the middle of September the activity seem to decrease again until the end of the analyzed time period.

6.1.5   Discussion

Both preprocessing reduced color variation (Figure 6.6) and camera movement. Nevertheless, they did not solve the color changes due to varying in-optical properties [Ioccg, 2000; Johnsen et al., 2009]. Events such as plankton blooms or re-suspension of sediments highly affect image color (Figure 6.8). For the coral color monitoring these changes can be normalized during the coral color feature extraction (Section 6.1.2 Image processing for color monitoring). For the PA monitoring, different image preprocessing for underwater images (Chapter 5) were tested to compensate for these color-shifts, but all resulted in lower classification performance (not shown). The image alignment produced reliable results even for such a long period. Working with live organisms adds more challenges to the image alignment as a growth of coral branches can be noticed throughout the investigated time period.

Coral segmentation using a fixed extraction mask $\hat{\mathbf{M}}^{(\mathrm{coral})}$ allowed to keep the segmentation area even for images with very turbid water. Changes caused by the DOP challenge have no influence on the coral segmentation.

The selected and extracted feature $\widetilde{c}_{t,a*}^{(\mathrm{coral})}$ is suitable to monitor the color change of the cold water coral and allowed to identify a possible seasonal long-term change. The use of a fixed mask guaranties that the analyzed parts of coral will not change over time. Hence, estimating a color shift only due to analyzing different coral parts over time is excluded. Short-term events, like fish occurrence, produce significant outliers only on for $\bar{c}_{t,a*}^{\mathrm{full}}$ (a*-channel of average color of the full image). Identification of those is a straightforward process and could also be automated using outlier detection e.g. as described in [Chen et al., 1993]. Nevertheless, the integration of sensor data like turbidity, or chlorophyll measurements in either the image processing or later analysis could improve the robustness of the estimated color changes further.

PA monitoring required generating manual annotations. Generating manual annotations is a dynamic process, which needs to be planned and carried out with care (Chapter 3). Annotation classes need to be defined at an early stage during this process. A preliminary study helped to define those classes and to calibrate them among all observers. Nevertheless, the annotation task is challenging using the defined six classes, as demonstrated by low inter- and intra- observer agreements. Merging the annotation classes allowed to increase the reproducibility and thereby the reliability of the annotations at the cost of a reduced level of detail.

The creation of a gold standard produced annotations with higher quality and reliability. Many annotations have been removed due to no or low overlap or class differences, as the matching was carried out using the six annotation classes.

The deep convolutional network, trained for the automated PA estimation, performed very well on the training, validation and test patches. The recall, estimated on $\bar{\mathbf{r}}^{(j)}$ to evaluate the performance on a pixel level, might give the impression of a low performance, but it needs to be interpreted considering the false positive rate, too. The κ values (Equation 6.5) for $g^{(j)}$ with $\omega^{(j)} =$ "active" (recall) differ significantly (highest significance was estimated using ANOVA) from κ values for $g^{(j)}$ with $\omega^{(j)} =$ "retracted" (FPR). Additionally, κ values were estimated using all annotation classes and the gradient of PA is also reflected by the estimated κ values for the corresponding classes (not shown).

The good performance of the PA estimation system evaluated on the full annotation images $\{\tilde{\mathbf{I}}_{t'}'\}$ is illustrated in Figure 6.9c. Looking at images with low $\varphi_{t'}$ (Equation 6.8), where most polyps were annotated as "retracted", $\theta_{t'}$ is lower than $\varphi_{t'}$. This might be due to the fact that many polyps in these images show a very low activity. Polyps with a higher activity will therefore be more likely identified and annotated by human observers. Looking at images with a higher $\varphi_{t'}$ (most polyps were annotated as "active"), $\theta_{t'}$ is higher than $\varphi_{t'}$. An explanation for the latter effect might be that human observers tend to underestimate the activity, whereas they will likely oversee some active polyps in images of high PA.

Most important, the automatically classified pixel-wise PA rang-correlates with the annotations (Figure 6.9c). So even if θ_t does not map perfectly with the manual annotations, a qualitative statement for the PA can be made.

The PA estimation failure during the algae bloom period is noticeable. Events where particles impact the visibility of the polyps will always have an impact on the detectability even for human observers. It can be expected that a preprocessing of the images does not solve

this challenges. Including sensor data from a turbidity meter of the LoVe observatory in the analysis will improve the interpretation of estimated activity.

6.1.6   Conclusion

Investigating the behavior of sessile species in images recorded with a fixed camera frame simplifies the CV approach. After aligning recorded images the locations of the OOIs are fixed and the detection is simplified. The area of interest can be identified using a CV assisted method as done for the coral color monitoring (coral mask $\hat{\mathbf{M}}^{(\mathrm{coral})}$) or by a manual selection as done for the estimation of the PA (polyp classification mask $\mathbf{M}^{\mathsf{polyps}}$). After this area of interest is identified, the computational task is reduced to a classification of the species state. The coral monitoring allowed to reveal an unexpected temporal change of color for the cold water coral *Lophelia pertusa* (Figure 6.8). The presented workflow of image processing and feature normalization allows to compare two different regions, i.e. micro-habitats, over long-time periods for color changes. It allows to identify and to quantify these changes.
The presented workflow for the PA monitoring allows to estimate the activity of *Lophelia pertusa* coral polyps over long time-periods with a temporal resolution of 1 image per hour. Expanding the analyzed time period to at least one year is of high interest, but not possible right now. Due to technical challenges the LoVe Ocean Observatory has not been operational since November 2015.

Short-term and long-term changes that were identified by the semi-automated workflows will be analyzed further in Chapter 8. Correlations between PA, coral color and sensor data acquired by additional sensors of the LoVe observatory will be examined.

6.2 Mobile species: shrimp

In contrast to sessile species (previous Section), mobile species are more difficult to monitor using images, either by manual annotation or by computational approaches. Possible multiple instances of species occurrences need to be detected and localized in the analyzed images to investigate the spatial and temporal distribution.

The here investigated mobile species are the shrimp. As in [Purser et al., 2013], no attempt is made here to identify shrimp down to species level. Reliable classification of shrimp requires additional physical sampling and no such sampling has been performed on the LoVe observatory, so far. Instances of shrimp are particularly hard to detect as they are semi-transparent and can be rather small (20×20 pixel) in the analyzed images. Furthermore, they can occur in high numbers with possible overlap in one image. The Shrimp is one group of organisms that are present in high numbers in Norwegian reefs [Purser et al., 2013]. Several shrimp species have been identified, dominated by *Pandalus* spp. [Jonsson et al., 2004; Mortensen et al., 2006; Purser et al., 2013]. Despite its documented presence, shrimp abundance across micro-habitats in cold water coral reefs have not been studied in detail [Purser et al., 2013].

While some work has been published about computational segmentation of cold water corals [Purser et al., 2009; Tusa et al., 2014], computational segmentation of the associated fauna is scarce. Existing studies of image analysis applied on shrimp or morphological related species [Harbitz, 2007; Gorsky et al., 2010; Benfield et al., 2007] have removed the studied species from their natural habitat before imaging to gain optimized imaging conditions. Computational *in-situ* detection of shrimp has been applied previously only in [Purser et al., 2013]. Shrimp eye reflection from the ROV (remotely operated underwater vehicle) lights in video images are used in this study, as a feature for the automated detection. This approach can not be applied here, as the shrimp eye contrast is not sufficient in the analyzed images.

As images were recorded by a stationary observatory, a different approach could be applied, here. Temporal Color Contrast Features (TCCF) are introduced. TCCF allow to use color for shrimp detection although shrimp are semi-transparent. This feature driven approach enables to handle the shrimp color variations. The TCCF also allow to compensate the OP challenge. Colors are processed in relation to their pixel position in the image and therefore also to the object-camera distance as this is stable between different images (after the image alignment). The method identifies and maps the distribution of shrimp in the micro-habitats of a cold water coral reef automatically. The computed shrimp abundance is visualized as a shrimp abundance map (SAM). OP challenge, DOP challenge -> TCCF AC challenge, AQ challenge, AU challenge

6.2.1 Material

For the purpose of shrimp monitoring using images from the LoVe observatory, a time period with a stable camera orientation had to be chosen. A stable camera orientation is precondition to apply TCCF. For this study the LoVe image set 1 is selected, featuring about 1.5 month
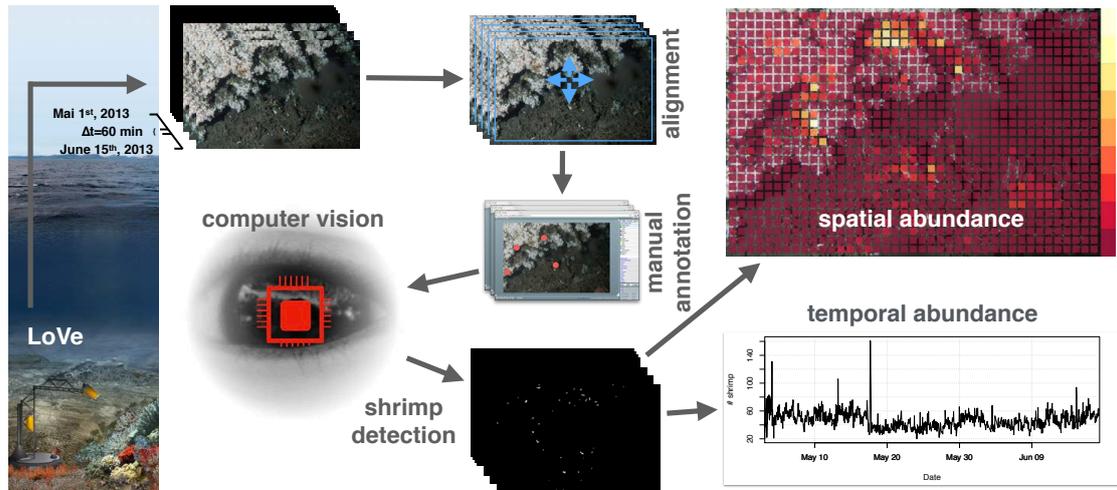
Figure 6.11: The workflow of the computational shrimp monitoring. Images recorded at the LoVe observatory are uploaded in a temporal resolution of one image per hour. Images are then spatially aligned (Section 6.2.2), followed by a manual annotation with the BIIGLE system (version 1.22), to generate shrimp examples required to apply ML. After training a random forest classifier (Section 6.2.2), shrimp instances are detected in all analyzed images (Section 6.2.2). The shrimp distribution is finally visualized in a scatter plot and a shrimp abundance map (Section 6.2.2). The latter visualizing temporal and spatial abundance.

of stable camera orientation. Compared to LoVe image set 2 the LoVe image set 1 the camera-reef distance is shorter allowing to locate and identify more shrimp. LoVe image set 1 contains 1140 images which were collected within 6 weeks (1st of May 2014 0:17:19 until 18th of June 2014 08:17:19). More details can be found in Section 4.2.1.

6.2.2   Methods

The different steps of the shrimp detection and location workflow, presented in Figure 6.11 are described in detail in the following. First, images are spatially aligned in the preprocessing to compensate small camera shifts. Second, a subset of 80 images are annotated manually for shrimp occurrence to provide data for training and testing of a ML algorithm. Third, temporal color contrast features (TCCF) are computed from a super-pixel representation of the original images. Using these TCCF, a supervised learning algorithm is trained on the manually annotated images. Finally, a pseudo color heat-map, referred to as shrimp abundance map (SAM), is derived from the positions found in all images to allow a first impression of the dynamics of shrimp positions.

Image Preprocessing   The image processing for the shrimp monitoring ($\mathbf{I}_t \mapsto \hat{\mathbf{I}}_t$) differs from the preprocessing of the previous Section 6.1. Here, the OP challenge is resolved by using TCCF (Paragraph Temporal color contrast features). No illumination or color correction

was applied, as the image collecting setup is stable in this regard over the total set of images. The power supply is not influenced by decreasing power of batteries and at 260 m no natural illumination exists. Effects of the DOP challenge were not considered, as a manual inspection showed only low variation for the rather short time period (about 6 weeks).

Here, the preprocessing consist therefore only of a spatial alignment of the images to compensate camera shifts caused by changing current speed and directions. The spatial alignment is necessary in order to apply the TCCF and therefore to use the high likeliness of shrimp movement between two images (i.e. one hour time lag) as a feature for discrimination between shrimp and not shrimp. In principle images can be aligned as done for the monitoring of the sessile species (Section 6.1.2), but due to historical reasons a different image alignment is applied, here. The monitoring for the sessile species required a more advanced image alignment as the analyzed time period is longer.

Here, all 1140 images ($w = 5184$ pixels, $h = 3456$ pixels) are aligned to the first image $\mathbf{I}_1$, the so called reference image, collected at 1st of May 2014 0:17:19. $\mathbf{I}_1$ is initially cropped by 50 pixels on all four image borders, allowing a shift of maximal 50 pixels in all four directions for the following images, the so called moving images. The alignment shifts are computed, using Canny edge filtering (Section 2.3.3) with a low threshold of 30 and an high threshold of 90. The result of the Canny filtering of an image $\mathbf{I}_t$ is denoted here as the Canny edge map $\mathbf{I}_t^{(E)}$. Each image $\mathbf{I}_t$ is transformed to a new aligned version $\hat{\mathbf{I}}_t$:

$$\hat{\mathbf{I}}_t = \Psi(\mathbf{I}_t, x_t^{(\text{best})}, y_t^{(\text{best})}) \tag{6.9}$$

using a transform function Ψ, i.e.

$$\Psi(\mathbf{I}_t, x', y')_{x,y} = \mathbf{I}_t(50 + x - x', 50 + y - y')$$

that i) reduces the width w and height h of the image ($\Psi(\mathbf{I}_t, x', y') \in \mathbb{R}^{(w-100)\times(h-100)}$, see cropping instructions above) and ii) shifts all pixels by the vector (x', y'). The best shifting parameters $(x_t^{(\text{best})}, y_t^{(\text{best})})$ are computed as

$$(x_t^{(\text{best})}, y_t^{(\text{best})}) = \operatorname*{argmax}_{(x', y') \in \{-50,...,50\}^2} \left\{ \sum_{x,y} (\Psi(\mathbf{I}_t^{(E)}, x', y'))_{x,y} \cdot (\Psi(\mathbf{I}_1^{(E)}, 0, 0))_{x,y} \right\}.$$

Manual annotations In order to provide necessary manual annotations for the applied ML, a small subset of $N = 80$ of the preprocessed images were uploaded to the BIIGLE 1.22 [Ontrup et al., 2009; Schoening et al., 2009] platform (Section 3.3). Using BIIGLE on these N images the positions of shrimp are manually marked with point annotations $a_t^{(j)}$ using a computer mouse. Here, each $a_t^{(j)}$ consist of a position (x, y) and only one annotation class $\omega = \text{shrimp}$:

$$a_t^{(j)} = ((x, y), \omega = \text{shrimp})$$

All $a_t^{(j)}$ are used for training and testing the computational shrimp detection algorithm (see Chapter 6.2.2). Annotations are publicly accessible via `https://ani.cebitec.uni-bielefeld.`

de/biigle/ with the username `shrimp` and the password `shrimp` under the area `love` and transect `May June Selection`. The annotators are allowed to place more than one point shaped annotations $a_t^{(j)}$ on each shrimp, so that the whole shrimp is covered by multiple annotations.

Temporal color contrast features   Shrimp are semi-transparent, which causes a color variation of the individual shrimp dependent of the background color. To enhance the contrast between shrimp and background additional information from previously recorded images is utilized. In order to utilize this in a feature representation at a pixel position (x, y), the TCCF as $\mathbf{I}_t^{(C)}$ is computed for each pixel in the CIELab color space [Schanda, 2007] by

$$\mathbf{I}_t^{(C)}(x, y) = \hat{\mathbf{I}}_t(x, y) - \mathbf{I}_t^{(\text{median})}(x, y)$$

with

$$\mathbf{I}_t^{(\text{median})}(x, y) = \hat{\mathbf{I}}_{m(x,y,t)}(x, y) \ . \tag{6.10}$$

The image index $m(x, y, t)$ gives the image pixel $\hat{\mathbf{I}}_{m_{(x,y)}}(x, y)$ for the median image and is computed as the arg median on the $L*$-channel at position (x, y) over the 50 previous images:

$$m(x, y, t) = \underset{t'=t-50,..,t}{\arg \text{ median}} \left(\left\{ \hat{\mathbf{I}}_{t'}^{(\text{L*channel})}(x, y) \right\} \right)$$

The median image $\mathbf{I}_t^{(\text{median})}$ is computed as a floating median in the CIELab color space, resulting in an image including only sessile objects and thereby an image without shrimp (Figure 6.12, lower left). $\mathbf{I}_t^{(\text{median})}$ is modeling the background, which is adapted over time as the median is a floating median. The result can be visualized as a contrast map $\mathbf{I}_t^{(C*)}$ (Figure 6.12, lower right) by computing

$$\mathbf{I}_t^{(C*)} = \frac{\hat{\mathbf{I}}_t - \mathbf{I}_t^{(\text{median})}}{2} + 127 \ . \tag{6.11}$$

Super-Pixel Segmentation   All images $\hat{\mathbf{I}}_t$ are pre-segmented using the Simple Linear Iterative Clustering (SLIC) super-pixel algorithm [Achanta et al., 2012] (Section 2.7.1: Super-pixel). The algorithm is applied with a feature weight of 40 and a step size of 20 pixels. An image $\hat{\mathbf{I}}_t$ consists therefore of several super pixels $\{s_t^{(i)}\}$ $(|\{s_t^{(i)}\}| \approx 40,000)$, which are groups of locally connected pixels, as shown in Figure 6.13. The super-pixel segmentation is used to reduce the number of feature extraction positions. Instead of extracting features on e.g. overlapping quadratic regions, image information of local pixel similarities is used to find regions for feature extractions. The super-pixel size was optimized to cover the size variation of the shrimp, resulting in a size of one super-pixel for the smallest shrimp.
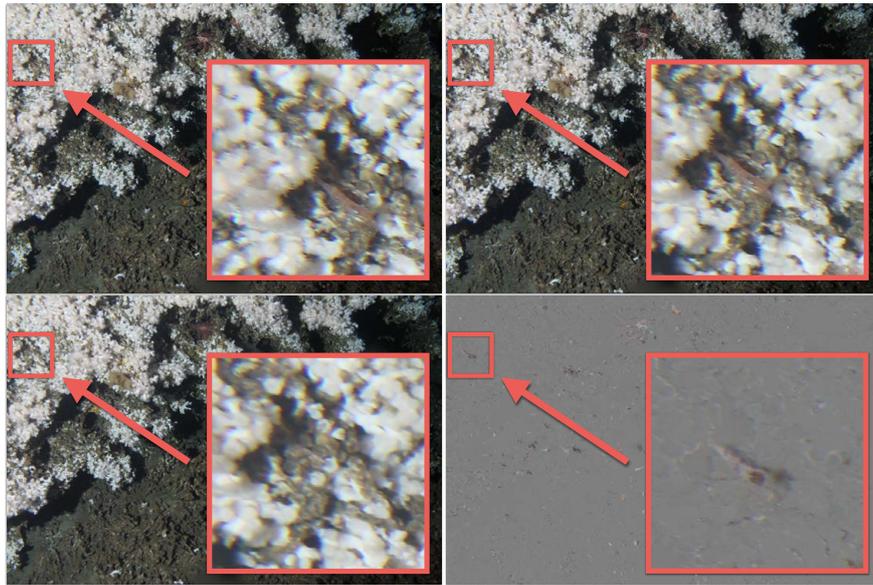
Figure 6.12: Visualization of the preprocessing steps from image to the shrimp contrast map. In the upper left an image $\mathbf{I}_t$ form the image time-series can be found. In the upper right the corresponding aligned image $\hat{\mathbf{I}}_t$ (see Equation 6.9) is displayed, which is slightly shifted. In the lower left the median image $\hat{\mathbf{I}}_t$ (see Equation 6.10) is shown, where no moving objects like the shrimp can be found. In the lower right the shrimp contrast map $\mathbf{I}_t^{(C^*)}$ (see Equation 6.11) is displayed.
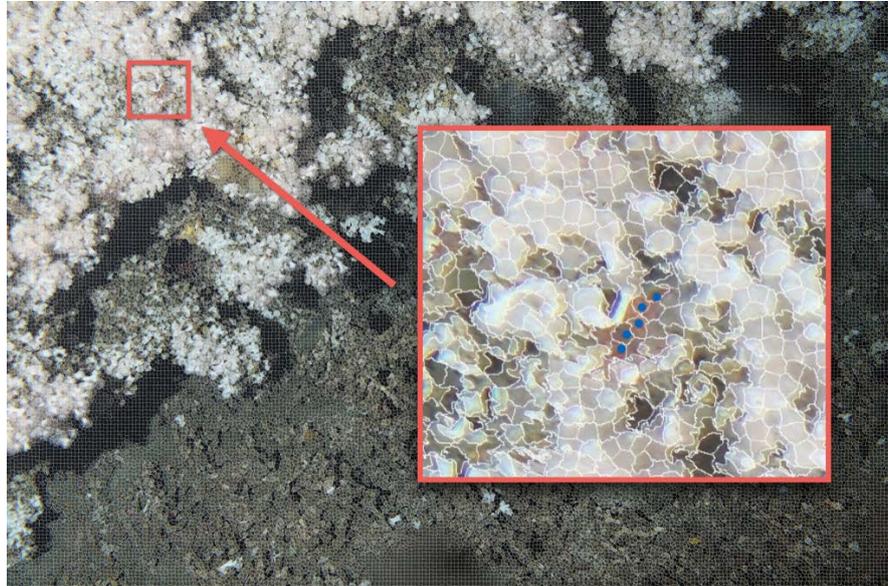
Figure 6.13: The applied SLIC (Simple Linear Iterative Clustering) super-pixel segmentation. White contours outline the super-pixel areas. In the zoomed area examples for annotations of a shrimp (blue color) can be found, visualizing multiple annotations per shrimp.

Table 6.1: Description of the features. Features are extracted on the preprocessed images $\hat{\mathbf{I}}$ and on the shrimp contrast map $\mathbf{I}^{(C)}$. The feature consist of mean values and 16 bin histogram values computed in the CIE Lab color space (Section 2.2.2).

$\overline{L_{\hat{\mathbf{I}}}}, \overline{a_{\hat{\mathbf{I}}}}, \overline{b_{\hat{\mathbf{I}}}}$	:	mean values computed from $\hat{\mathbf{I}}$
$h_{0,L,\hat{\mathbf{I}}}, \cdots, h_{15,L,\hat{\mathbf{I}}}, \cdots, h_{15,b,\hat{\mathbf{I}}}$	:	16 bin histogram values computed from $\hat{\mathbf{I}}$
$\overline{L_{\mathbf{I}^{(C)}}}, \overline{a_{\mathbf{I}^{(C)}}}, \overline{b_{\mathbf{I}^{(C)}}}$	:	mean values computed from $\mathbf{I}^{(C)}$
$h_{0,L,\mathbf{I}^{(C)}}, \cdots, h_{15,L,\mathbf{I}^{(C)}}, \cdots, h_{15,b,\mathbf{I}^{(C)}}$	:	16 bin histogram values computed from $\mathbf{I}^{(C)}$

Feature Extraction   Features are extracted for each super-pixel $s_t^{(i)}$ on $\hat{\mathbf{I}}_t$ and corresponding $\mathbf{I}_t^{(C)}$ represented by a feature vector $\mathbf{v}_t^{(i)} \in \mathbb{R}^{38}$. Extracted features are mean values and histogram values (16 bins) in the CIELab color space (see Table 6.1). A class $\omega_t^{(i)} \in \{1 = \text{shrimp}, -1 = \text{not shrimp}\}$ is assigned to each feature vector $\mathbf{v}_t^{(i)}$ referring to the corresponding super-pixel $s_t^{(i)}$. All 38-dimensional $\mathbf{v}_t^{(i)}$ are thus either labeled as shrimp if one ore more $a_t^{(j)}$ were placed in the area of the corresponding $s_t^{(i)}$ or not shrimp otherwise.

Machine Learning   For the automated detection of shrimp, the Random Forest (RF) [Breiman, 2001] algorithm, which is a supervised ML algorithm (Section 2.7.2), was selected. The algorithm allows to use a smaller training set, compared to e.g. the Deep Learning algorithms like CNNs (Section 2.7.3) and allows training with a highly unbalanced training set. The latter is important, as the strategy is to use all not-annotated super-pixel as training examples for the not shrimp class resulting in a highly unbalanced set. Furthermore, it is very robust against over-fitting and can also be used to compute the feature importance (Figure 6.20). The RF is trained and tuned using a four-fold cross validation (Section 2.8 for details). Therefore, the set of 80 annotated images is divided into four different sets of training and test images. The RF is trained on the labeled feature vectors of the training set of images $\{(\mathbf{v}_t^{(i)}, \omega_t^{(i)})\}_{\text{train}}$. The training parameters of the RF are tuned on feature vectors and their corresponding label belonging to the test set of images and were varied for the maximum tree-depth $(5, 10, 20, 50, 100, 150)$, the maximum number of trees in the forest $(100, 500)$ and the minimum sample count $(5, 10, 20)$. The class-weights for the shrimp class were also varied $(2, 5, 10, 100)$. Other parameters were not varied, resolving in a training of in total 144 different RFs.

Post-Processing   To optimize the classification performance the classification output (Δ) is post-processed using a size criterion. Connected super-pixels that are all classified to be shrimp, form a shrimp region $\mathbf{r}_t^{(j)}$ of size $|\mathbf{r}_t^{(j)}|$. In contrast to the previous chapter, $|\mathbf{r}^{(j)}|$ counts the number of $s_t^{(i)} \in \mathbf{r}_t^{(j)}$. All $\mathbf{r}_t^{(j)}$ are filtered to be in range of $1 < |\mathbf{r}_t^{(j)}| < 11$, excluding very small $(|\mathbf{r}_t^{(j)}| = 1)$ and very large regions $(|\mathbf{r}_t^{(j)}| > 11)$ of shrimp classifications

(a) $|\mathbf{r}| = 1$ (b) $|\mathbf{r}| = 5$ (c) $|\mathbf{r}| = 16$
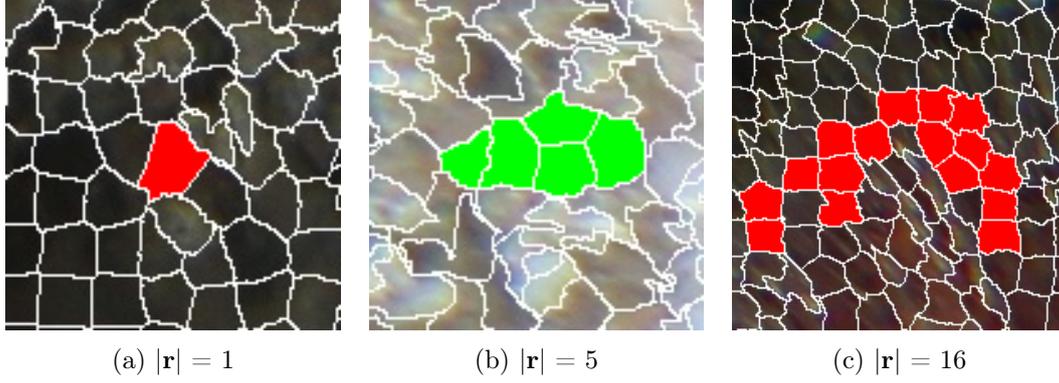
Figure 6.14: The post processing of the classification outputs $\Delta(\mathbf{v}_t^{(i)})$ of the trained Random Forest (RF). (a) has a super-pixel size of one ($|\mathbf{r}| = 1$) and will therefore be excluded, (b) has a super-pixel size of five ($|\mathbf{r}| = 5$) and therefore fulfills the size criteria while (c) will be excluded as it is to big ($|\mathbf{r}| = 16$).

(Figure 6.14).

Shrimp Abundance Map   The shrimp abundance map (SAM) visualizes the spatial distribution of the shrimp over a defined time-period T. The cold water coral reef scene is divided into a grid of equal sized cells $q^{(n)}$ with w = 120 and h = 120. For all cells $q^{(n)}$ the relative shrimp abundance is visualized using a heated object color-scale with 9 levels (9-class `YlOrRd` [Brewer, 2013; Brewer, 1999]). The color scale is starting at dark red (RGB color code: 128,0,38) for the lowest abundance and ending at a light yellow (RGB color code: 255,255,204) for the highest abundance (e.g. Figure 6.18, on the right). The relative shrimp abundance is computed as

$$\alpha^{(n)} = \log_2 \left(\frac{\tau^{(n)} - \min_{n'}(\tau^{(n')})}{\max_{n'}(\tau^{(n')}) - \min_{n'}(\tau^{(n')})} + 1 \right),$$ (6.12)

where $\tau^{(n)}$ is the shrimp abundance for the cell $q^{(n)}$

$$\tau^{(n)} = \sum_{t \in T} H \left(\sum_{s_t^{(i)} \in q^{(n)}} \Delta(\mathbf{v}_t^{(i)}) \right)$$

where $s_t^{(i)} \in q^{(n)} <=> s_t^{(i)} \cap q^{(n)} \neq \{\}$ and with H is defined similar to the *Heaviside* step function:

$$H(x) = \begin{cases} 0: & x = 0 \\ 1: & x > 0 \end{cases}.$$

An average image of the images of the visualized time-period is used as a background image (Figure 6.18).

Evaluation   The evaluation of an image analysis system's accuracy can be a complex subject. In general manual annotations of image samples are compared with the corresponding classification outputs $\Delta(\mathbf{v}^{(i)})$ and grouped into true positives (TPs), false positives (FPs), false negatives (FNs) and true negatives (TN) (Section 2.10). In this two-class classification problem, here $\omega_t^{(i)} \in \{1 = \text{shrimp}, -1 = \text{not shrimp}\}$, they are defined as:

$$\text{TP} = |\Delta(\mathbf{v}_t^{(i)}) = 1 \wedge \omega_t^{(i)} = 1|,$$

$$\text{FP} = |\Delta(\mathbf{v}_t^{(i)}) = 1 \wedge \omega_t^{(i)} = 0|,$$

$$\text{FN} = |\Delta(\mathbf{v}_t^{(i)}) = 0 \wedge \omega_t^{(i)} = 1|.$$

As stated earlier, all $\mathbf{v}_t^{(i)}$ with no corresponding $a_t^{(j)}$ have the class $\omega_t^{(i)} = 0$. True negatives are therefore not considered as no manual negatives (i.e. not shrimp) were annotated explicitly.

In order to evaluate the performance of the trained RF two binary image mask for each $\hat{\mathbf{I}}_t$ are computed. The first binary image mask $\mathbf{B}_t^{(\omega)}$ is created by marking the pixel positions $\{(x,y) \in \{s_t^{(i)} | \omega_t^{(i)} = 1\}\}$ of the corresponding $\hat{\mathbf{I}}_t$:

$$\mathbf{B}_t^{(\omega)}(x,y) = \left\{ \begin{array}{ll} 1: & \exists i \, |(x,y) \in s_t^{(i)} \wedge \omega_t^{(i)} = 1 \\ 0: & \text{else} \end{array} \right. .$$

The second binary image mask $\mathbf{B}_t^{(\Delta)}$ is created marking the super-pixel $\{s_t^{(i)} | \Delta(\mathbf{v}_t^{(i)}) = 1\}$ of the corresponding $\hat{\mathbf{I}}_t$:

$$\mathbf{B}_t^{(\Delta)}(x,y) = \left\{ \begin{array}{ll} 1: & \exists i \, |(x,y) \in s_t^{(i)} \wedge \Delta(\mathbf{v}_t^{(i)}) = 1 \\ 0: & \text{else} \end{array} \right. .$$

Although using the super-pixel clustering the spatial extent for the annotations and classifications are increased, an additional spatial uncertainty is allowed by applying an image-dilation function ϕ to all $\mathbf{B}_t^{(\Delta)}$. In order to determine the TP, FP and FN counts, $\mathbf{B}_t^{(\omega)}$ and $\phi(\mathbf{B}_t^{(\Delta)})$ are compared region wise. Each group of connected pixels marked in $\phi(\mathbf{B}_t^{(\Delta)})$ is counted as a TP if parts of it are also marked in $\mathbf{B}_t^{(\omega)}$ and as a FP if not. All regions only marked in $\mathbf{B}_t^{(\omega)}$ are counted as FN. By applying the image-dilation function ϕ, non overlapping but touching regions in $\mathbf{B}_t^{(\Delta)}$ and $\mathbf{B}_t^{(\omega)}$ are therefore also counted as TPs. In Figure 6.15 examples for TP, FP and FN are visualized.

TP, FP, and FN are finally used to calculate the precision (Equation 2.3) and the recall (Equation 2.4) for the shrimp classification system as described in Section 2.10. The precision is measuring the relative number of correctly identified shrimp over all shrimp detections. The recall is measuring how many of the annotated shrimp are correctly identified to be shrimp by the RF. The F_1 score (Equation 2.6) is computed and used to optimize the parameters of the classification system. Precision and recall are weighted equally, i.e. the harmonic mean between both. Based on the assumption that the ground truth is correct, i.e. no shrimp are overseen and no non-shrimp area is labeled as shrimp, the best accuracy would be a precision of 1 and a recall of 1 resulting in an F_1 score of 1.
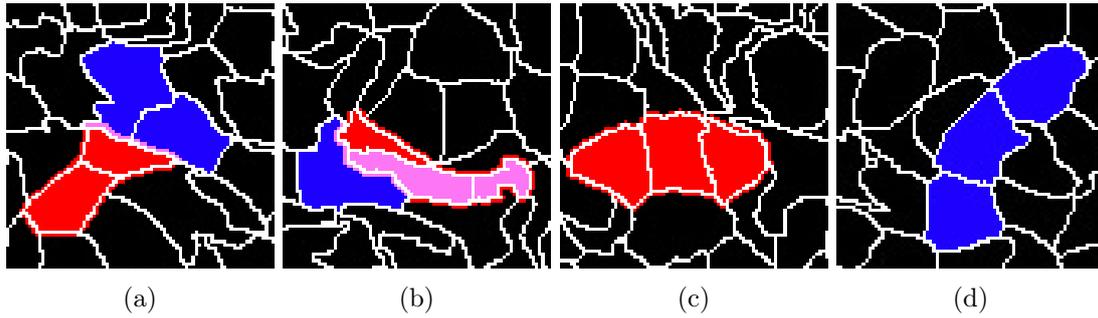
Figure 6.15: The evaluation of the performance of the random forest on entire images. Image patches showing the overlay of $\mathbf{B}_t^{(\omega)}$ and $\phi(\mathbf{B}_t^{(\Delta)})$ are presented for TP, FP and FN. White contours illustrate the super-pixel areas. Pixel values are coded in red for pixels only marked in $\phi(\mathbf{B}_t^{(\Delta)})$, in blue for pixels only marked in $\mathbf{B}_t^{(\omega)}$ and in pink for pixels marked in both. The effect of the dilation function ϕ is visible as a small boarder around $\{s_t^{(i)}|\Delta(\mathbf{v}_t^{(i)}) = 1\}$ marked in red. (a), (b) are displaying both TPs. Please note that (b) is counted as a TP due to the dilation rule (see text for details). (c) is displaying a false positive (FP) and (d) a false negative (FN).

6.2.3 Results

The results of previous described methods are presented in the following. The image preprocessing consist only of an image alignment, which was evaluated by visual inspection of a rendered time-lapse video. This showed that the image alignment process reduced the shift between the images.

The annotators were allowed to place multiple $a^{(j)}$ on a shrimp, so that many shrimp are represented by a small cloud of marked points (Figure 6.13). In total $|\{a^{(j)}\}| = 3459$ annotations were placed on the 80 images. In a previously performed experiment the intra observer agreement (see Section 3.2.3) of one annotator was estimated. The annotator had to re-label 20 images after a time gap of four days, resulting in an agreement of 0.66 estimated on in total 3503 shrimp point annotations. This low agreement again motivates the need for an automation of the detection. Nevertheless, it indicates that shrimp can be identified in the images, as otherwise this agreement would have been much lower.

The super-pixel segmentation was evaluated manually using visual inspection. Most of the shrimp were segmented into several super-pixels (Figure 6.13) and only a few very small shrimp were covered by only one super-pixel. In a re-inspection of these one super-pixel sized shrimp it could be observed that such annotation show a low level of reproducibility. Shrimp positions were marked in the training set to consider this phenomenon in the following evaluation of the shrimp detection.

The evaluation of the accuracy of the shrimp detection (Section 6.2.2) was applied to optimize the training parameters of the RF to give the highest F_1 score. In Table 6.2 averages for precision, recall and F_1 score of the four-fold cross validation for the best RF parameters are given. The best F_1 score was achieved with a tree-depth of 50, a minimal sample count of

Table 6.2: The parameters of the Random Forest classifier are optimized using a stratified four-fold cross validation. The results averaged of the four-fold cross validation for training and test image sets are given here. The manual re-evaluation was performed for the test data only using a conservative ground truth (see text for details). Performances were evaluated by computing recall, precision and F_1 score (Section 2.10).

	Precision	**Recall**	F_1 **score**
Training	0.48	1	0.64
Test	0.69	0.52	0.59
Re-ev. test	0.77	0.91	0.84

20, 10 active variables, a weight of 10 for the shrimp class ($\omega = 1$), a maximal number of trees of 100 and an out of bag error of 0.001 (the last two both used as a termination criteria during the training).

Manual annotation can be very error-prone [Schoening et al., 2012; MacLeod et al., 2010]. Some shrimp will likely be overseen during the manual annotation process, as it is expected that the performance of the annotator will vary, as reported in the intra-observer study (see Section 6.2.1). The classification outputs were therefore manually re-evaluated region wise (Figure 6.16). Since a number of small shrimp showed a rather low reproducibility, computational results were re-evaluated with a second, more conservative ground truth: connected regions in $\mathbf{B}_t^{(\omega)}$ were excluded if they only consist of one super-pixel (Figure 6.16). This resulted in an improved F_1 score of 0.84 (see Table 6.2).

The number of detected shrimp in each image over the analyzed time period is displayed in the chart of Figure 6.17. The locations of the automated detected shrimp in each image were used the generate shrimp abundance maps (SAMs) for different time scales and periods. Figure 6.18 displays the relative shrimp abundance of all 1090 analyzed images (3rd of May 2014 02:17:19 until 18th of June 2014 08:17:19). In Figure 6.19a a SAM obtained by analyzing images recorded at day time between 8 am and 8 pm is compared with a SAM obtained by analyzing images recorded at night time between 8 pm and 8 am. In Figure 6.19b a SAM for the images from 11th of May is compared with the a SAM for the images from 18th of May. Figure 6.19c displays a comparison of the first half of the images (first 3 weeks, 3rd of May 2014 02:17:19 until 25th of May 2014 21:17:19) with the second half of the images (following 3 weeks, 25 of May 2014 22:17:19 until 18th of June 2014 08:17:19).

6.2.4 Discussion

In the preprocessing the analyzed images were aligned. Even though the proposed method does only correct for shifts in x and y direction, alignments were precise enough to compute TCCF.

The application of the super-pixel algorithm for pre-segmentation accelerated the automated

detection and provided a reasonable criterion ($1 < |\mathbf{r}^{(j)}| < 11$) to filter both manual annotations and automated classifications.

The use of point shapes instead of more complex shapes for the manual annotation reduced the time needed for the manual annotation with the cost of an increased spatial uncertainty of the ground truth for evaluating the classification system (AU challenge). The exact pixel location of such a point shaped annotation is more or less random in a certain extent across the object of interest (OOI), i.e. the shrimp in this case. In principle, this spatial uncertainty is not a problem for the later application of the CV system as for the purpose of identifying the spatial and temporal distribution of shrimp, it is sufficient enough to detect only parts of a shrimp, as long as for each shrimp a part e.g. the tail is detected. It only makes the evaluation a bit more complex. The increased spatial uncertainty can cause an underestimation of performance of a tested classification system, as $\Delta(\mathbf{v}_t^{(i)}) = 1$ can be counted as a FP and $\Delta(\mathbf{v}_t^{(i)}) = 0$ can be counted as a FN, although both are located on the same OOI, but not on the exact same pixel location. A solution can be to allow spatial differences between annotations and classification as it is done in [Schoening et al., 2012]. However, in case of very small OOI like shrimp this approach bares the potential of an assignment problem for close shrimp. In this approach $a_t^{(j)}$ are assigned to super-pixels $s_t^{(i)}$, representing a pixel area, which extended the point annotation to a super-pixel annotation with the class label $\omega_t^{(i)}$ which reduces the previous mentioned location problem. Furthermore, due to the annotation rule allowing to place multiple $a^{(j)}$ on each shrimp, a shrimp-region can consist of several super-pixels s^i with $\omega_t^{(i)} = 1$ forming a set of annotated super-pixels. The idea behind this is to benefit from the speed up of using point annotations and at the same time generate annotations with a meaningful spatial extend. Meaningful as, the super-pixel algorithm groups local similar colors.

Using the temporal color contrast feature enabled to discriminate between shrimp and not shrimp, as it combines information from previous recorded images with the current image. This improvement of the ability was validated using the method for ranking image preprocessing (Section 5.3.1) and by evaluating the feature importance of the best RF (total importance of 0.59 for features computed on $\mathbf{I}^{(C)}$ and total importance of 0.41 for features computed on $\hat{\mathbf{I}}$) (Figure 6.20). However, features computed on $\hat{\mathbf{I}}$ were not excluded, since it would have lowered the accuracy, as their importance was still high.

The F_1 score of 0.84, estimated in the manual re-evaluation, confirms that presented shrimp monitoring system produces reliable classifications. During the manual re-evaluation it was also observed that most of the remaining FPs were parts of relatively big objects like legs and body parts of crabs, parts of sea urchins, sea lilies and basket stars.
The detection sensitivity of the system is dependent on local image quality despite the use of the TCCF. Non or low illuminated areas of the images like holes or shadows, caused by the multi-spherical structures of the coral reef, leads to a lower detection sensitivity. However, as long as the camera position is fixed, it can be presumed that this effect is constant over time. Obviously, the dark spot areas caused by the multi-spherical structure and light settings, must be handled with care, if someone wants to compare results from different FUOs.

The presented system computes shrimp abundance charts for evaluation of the number of visible shrimp per image and SAMs for visualization of the relative spatial abundance in the same images over time. Qualitative differences and similarities can be recognized in an intuitive way e.g. in Figure 6.19b and Figure 6.19a, respectively. The visual differences between the relative shrimp abundance of the first three weeks and the following three weeks (Figure 6.19c), noticed during a manual inspection, were caused by some FP classification of sea lilies, which moved in the field of view in the second time period. Therefore, in the next development stage of the system, a pre-detection of big objects (in relation to the shrimp) as for instance the one presented in [Möller et al., 2016] is planed.

Impacts of DOP challenge have not explicitly been considered in this study. Although no such changes were recognized during the analyzed time period, it should be considered in the preprocessing of an improved system for longer time periods. This can also include an exclusion of images with very low visibility, e.g. caused by high turbidity.

The results from the present study show predominantly higher shrimp abundance in the live coral parts compare to the coral rubble or sediment parts of the image. This distribution corresponds to the findings of the study in [Purser et al., 2013].

6.2.5   Conclusion

The imaging setup, a fixed camera with a stable image frame, allows the use of the presented TCCF for investing the behavior of mobile species, in this case shrimp. TCCF could also be applied for detecting slower moving species, adapting the computation of the floating median image $\hat{\mathbf{I}}_t$ (modeled dynamic background) to the speed of movement. The only regulation here is that the investigated species moves faster than the modeled background. Furthermore, TCCF solve the DOP challenge as colors are processed depending on their position in the image and therefore individually for different camera-object distances.

The relative spatial abundance of shrimp over time can be visualized in SAMs and allows the identification of behavioral changes in an intuitive way. The shrimp distribution will be investigated in detail in future studies, following statistical analysis of the manual annotation sensitivity and automated detection performances on both areas. Although the investigated time period is short (about 6 weeks) changes in the temporal and spatial-temporal distribution could be identified. Investigating the shrimp behavior on the same image set LoVe image set 2 as for the sessile species and therefore on a longer time period, of course is of high interest. Due to the changed distance between coral reef and camera of the latter, a direct adaption of the proposed method is not possible. A concentration on bigger instances of the shrimp would be necessary. New manual annotations are required and a re-tuning of the different parameters is advised. As LoVe image set 2 features a longer time interval and an increased color channel depth, image preprocessing also needs to be adapted. Applying those modification, it can be expected to achieve a similar performance on LoVe image set 2 instances and similar images recorded in the future of other FUOs with similar parameters.
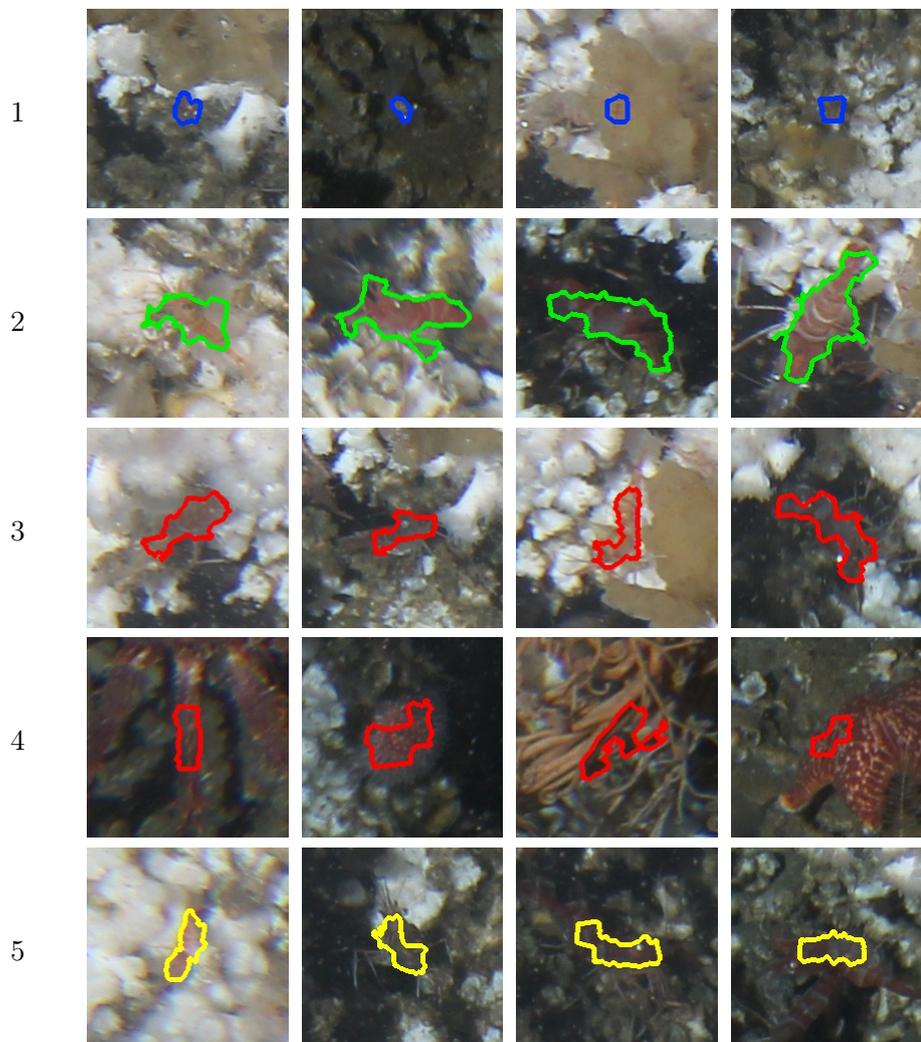
Figure 6.16: A selection of re-evaluated classification outputs, displaying in row 1 annotations (outlined in blue) that were excluded ($|\mathbf{r}^{(j)}| > 1$), in row 2 true positives (outlined in green), in row 3 false positives which are actually true positives (outlined in red), in line 4 false positives (outlined in red) and in row 5 false negatives (outlined in yellow).
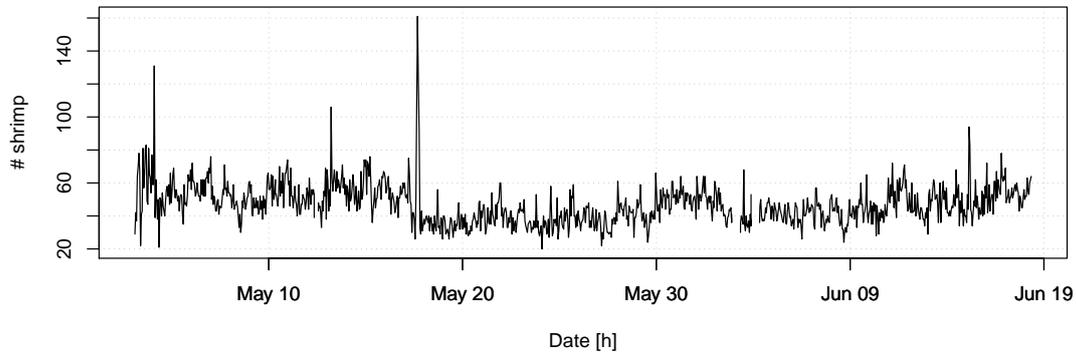
Figure 6.17: The Shrimp occurrence chart displays the number of detected shrimp (# shrimp) per image over the time period from early May (3rd of May 2014 02:17:19) until middle of June 2014 (18th of June 2014 08:17:19). The number of shrimp is counted as the number of connected shrimp regions in $\mathbf{B}_t^{(\Delta)}$ (see Section 6.2.2)
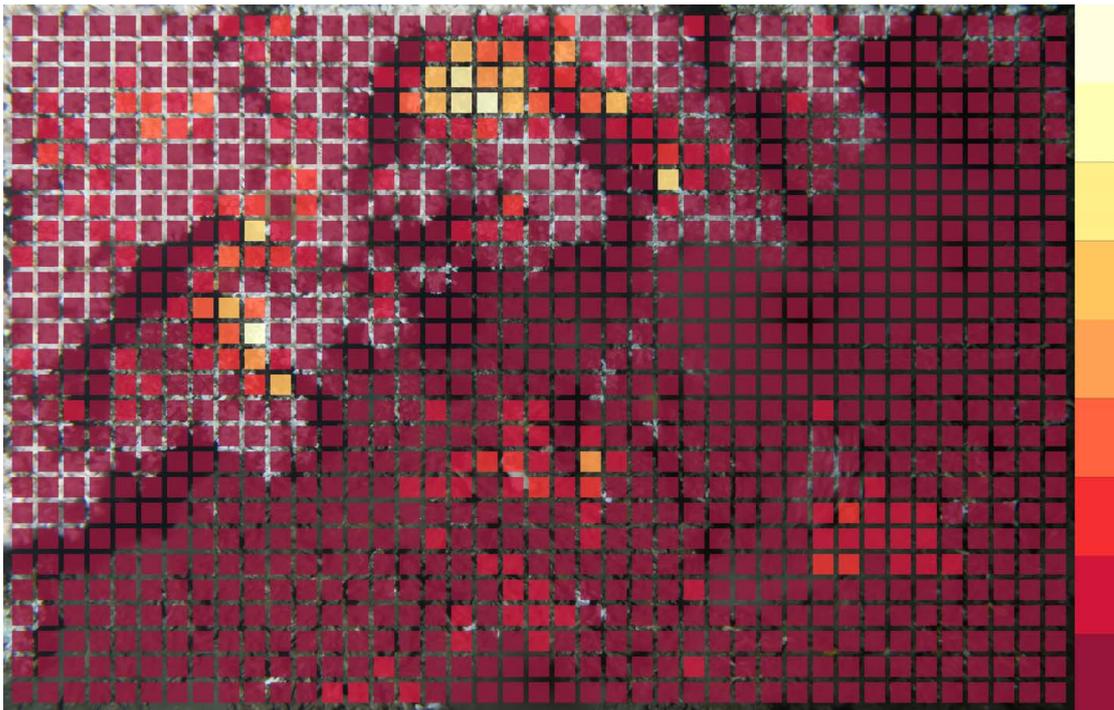
Figure 6.18: The shrimp abundance map (SAM) visualizing the spatial shrimp distribution from early May (3rd of May 2014 02:17:19) until middle of June 2014 (18th of June 2014 08:17:19). The color scale is displayed on the right starting with dark red for the lowest relative shrimp abundance $\alpha^{(n)}$ (Equation 6.12), continuing to red and orange indicating a higher relative shrimp abundance and ending at a light yellow for the highest relative abundance over time.
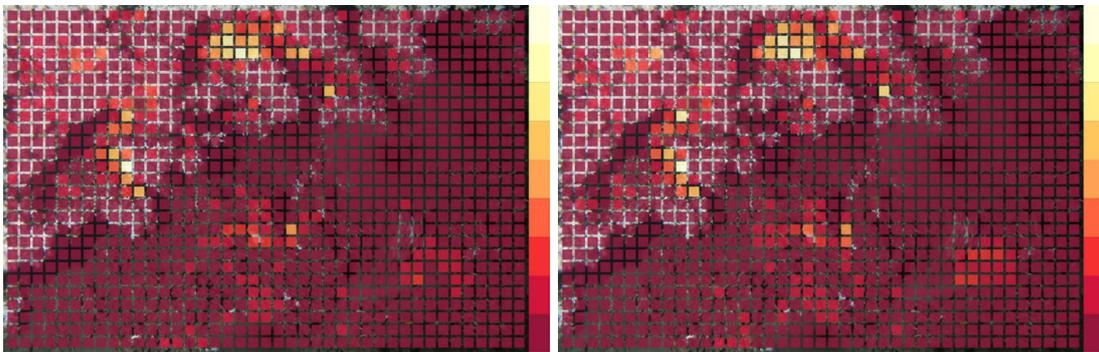
(a) Day (8 am - 8 pm) (left) vs. night (8 pm - 8 am) (right)
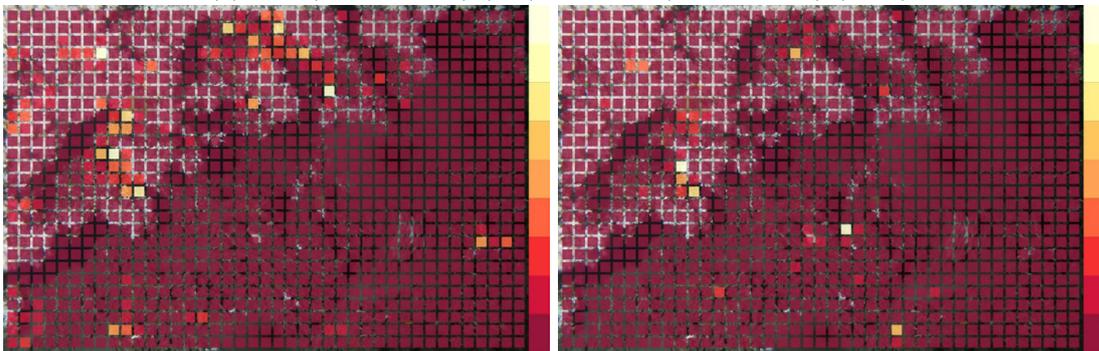
(b) 11$^{\text{th}}$ of May (left) vs. 18$^{\text{th}}$ of May (right)
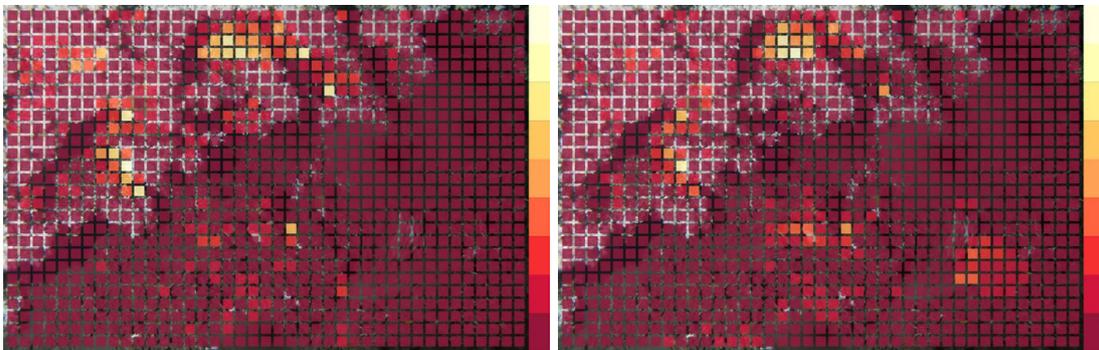
(c) First 3 weeks (3rd of May 2014 02:17:19 until 25th of May 2014 21:17:19) (left) vs. next three weeks (25 of May 2014 22:17:19 until 18th of June 2014 08:17:19) (right)

Figure 6.19: The shrimp abundance maps (SAMs) for the relative shrimp abundance for several time periods. In (a) the relative shrimp abundance at daytime is compared to the relative shrimp abundance at nighttime, in (b) the relative shrimp abundance of two different days is compared and in (c) the first half (three weeks) is compared with the second half (three weeks) of the analyzed time period.
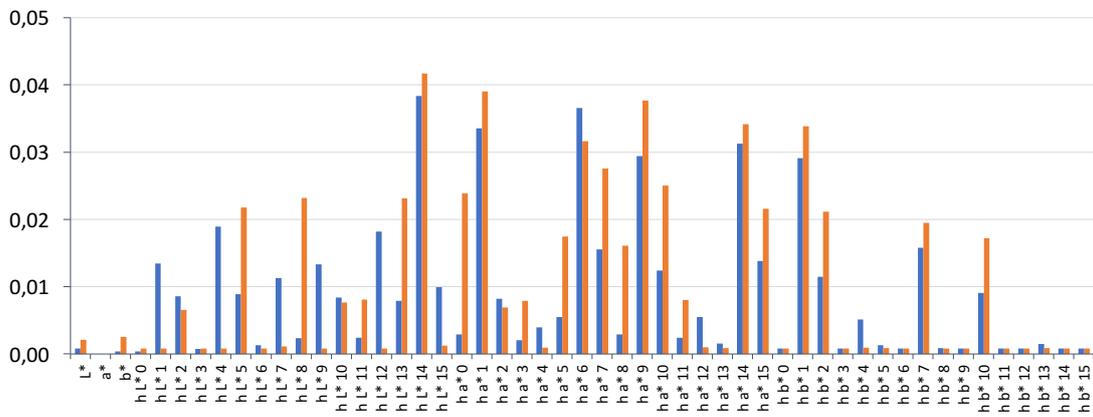
Figure 6.20: The feature importance for the best Random Forest. The importance for each variable is displayed, differentiating between features computed on $\hat{\mathbf{I}}$ colored in blue and features computed on $\mathbf{I}^{(C)}$ colored in orange. A slightly higher importance for features computed on $\mathbf{I}^{(C)}$ can be observed.

7 Computer vision for wet-lab experiments

In-situ monitoring is only one corner stone needed for a holistic environmental monitoring. In order to generate models for responses of the environment to anthropogenic activities, laboratory studies have to be conducted to identify key variables of the environment. Controlled conditions allow to investigate the influence of individual variables or combinations on the investigated species. In this section Computer Vision (CV) is applied to digital images collected from a laboratory experiment. In contrast to in-situ imaging, imaging conditions can be highly optimized for the specific purpose of the laboratory study. Nevertheless, unexpected difficulties can arise during the conduction of laboratory experiments, which have to be compensated in the later computational processing.

In this chapter, CV is applied to investigate the response of calcareous alga (CA), *Mesophyllum engelhartii* (Foslie) Adey to several treatment parameters. This section is based on publication 3.

7.1 Sessile species: Mesophyllum engelhartii

The *Mesophyllum engelhartii* (Foslie) Adey species belongs to the *Phylum Rhodophyta* under the Order *Hapalidiales*. Calcareous algae (CA) play an important ecological role in coastal habitats [Hall-Spencer, 1998; Steller et al., 2001; Riosmena-Rodríguez et al., 2010] contributing to the formation of rhodolites. Rhodolites are made from dead CA and other calcifying organisms [Tâmega et al., 2013]. These multi-spherical structures create a habitat for other organisms living on, between and in the structures [Basso, 1998]. CA are found down to about 250 meters water depth [Henriques et al., 2014; Littler et al., 1991; Littler et al., 1986] and the largest occurrence of rhodolite beds are found in the southwest Atlantic on the Brazilian continental shelf [Kempf, 1970; Foster, 2001; Amado-Filho et al., 2012]. These algae communities may be disturbed by natural sedimentation and/or sedimentation from anthropogenic activities such as fish-trawling, mining [Nelson, 2009] and discharges of drill cuttings from oil and gas drilling activities [Davies et al., 2007]

In order to investigate the impact from sedimentation of drill cuttings on live CA and to establish threshold levels, a laboratory study was conducted. The study was part of the "Peregrino Environmental Monitoring Calcareous Algae" project. Rhodolites, partly covered with CA, were collected from the Peregrino oil field off the coast of Brazil. The impact of sedimentation was investigated in a laboratory flow-through system. Investigated parameters were light exposure ($\mathcal{L}$), flow rate ($\mathcal{F}$), amount of sediment ($\mathcal{S}$) and time ($\mathcal{T}$). In order to analyze the response of the CA samples to the investigated parameters (predictor variables), sediment coverage (SC) and photosynthetic efficiency (P) was measured (response variables). The laboratory study is discussed and published in [Figueiredo et al., 2015]. The digital images, also recorded during the conduction of this experiment, are analyzed here. CV methods are applied and adapted in order to extract visual calcareous algae features, capable to quantify the response to the investigated parameters, from the recorded images.
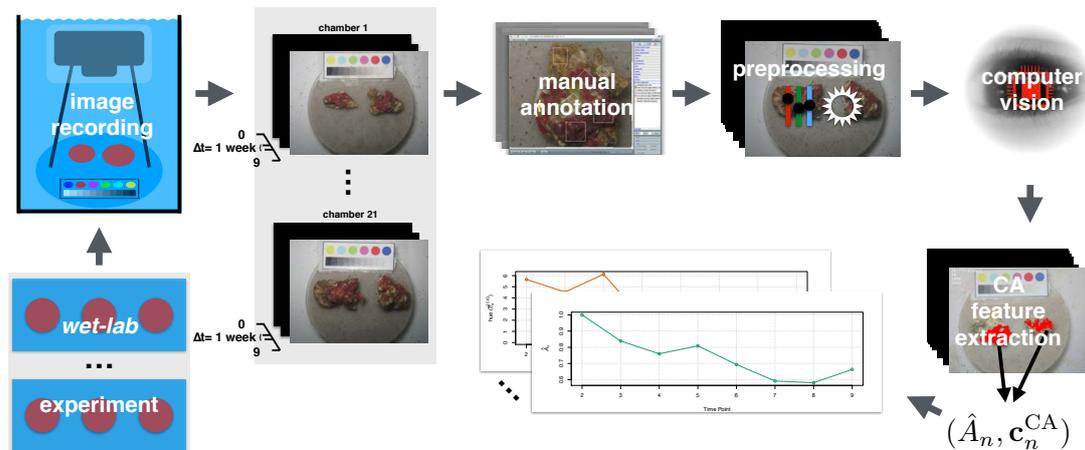
Figure 7.1: The workflow for the calcareous algae impact quantification. During the experiment conduction, CA samples are removed of their chambers for image recording weekly. All recorded images are manually annotated by biologist experts for live, stressed and dead CA with an example based annotation strategy. From now on the workflow does not include any manual interaction. Images are preprocessed for color constancy purposes. CV is applied for the automated segmentation of CA and color and size are extracted from the segmented CA to quantify its change related to the different treatment parameters.

These visual features are color and size change of the CA samples. Using images to record those features increases the transparency of the evaluation of the experiment. The results are encapsulated in images and which can be re-inspected.

A computational analysis of the recorded images is needed to speed up the process of color and size quantification. Furthermore, humans have a limited capability to quantify visual features, such as color change, in an objective way [Rensink et al., 2000; Wolfe et al., 2006]. The proposed CV workflow allows to quantify visual changes in form of size and color of the investigated species. After an initial manual annotation, this workflow allows to quantify color and size change fully automatically. To learn a classification function for the segmentation of CA, an H²SOM algorithm is applied (Section 2.7.1 Hierarchical Hyperbolic Self Organizing Map). The H²SOM has also been applied in Section 6.1.2 for segmenting cold water corals. Nevertheless, the integration of the basic algorithm into an image analysis pipeline needed to be modified regarding preprocessing, feature computation, pixel classification function and post-processing. A simplified overview of the whole impact quantification workflow is given in Figure 7.1.

7.1.1   Material

Laboratory experiment   Samples of live calcareous algae *Mesophyllum engelhartii* (Foslie) Adey were collected from $94 - 103$ m water depth at the Peregrino oil production field ($23°13'28.34''$ S, $41°50'55.73''$ W) located off the Brazilian Atlantic coast [Tâmega et al.,

2013]. The permission to collect the CA samples was granted by the Chico Mendes Institute for Biodiversity Conservation SISBIO license n 20820-2 and 20826-1. Light intensity ($\mathcal{L}$), flow rate ($\mathcal{F}$) and sediment amount ($\mathcal{S}$) could be varied independently as described in [Figueiredo et al., 2015]. In order to optimize the experiment, the three predictor variables ($\mathcal{L}$, $\mathcal{F}$, $\mathcal{S}$) were combined according to a Central Composite Face (CCF) design with center point [Box et al., 1978]. This enables regression modeling to describe linear and non-linear relationships and interactions between the variables. The full factorial design gives 15 combinations of $\mathcal{L}$, $\mathcal{F}$, and $\mathcal{S}$. Two of these combinations were duplicated and two were triplicated (i.e. technical replicates). Using the 15 combinations and technical replicates resulted in a total of 21 experiments which were carried out in 21 chambers (Table 7.1). In addition controls were kept in complete darkness and without sediment. The area of the chamber bottom was approximately 0.06 m^2 and the height of it was 0.1 m resulting in a volume of approximately 6 L. The light levels ($\mathcal{L}$) used in the exposure studies were 3, 6.6, or 10 $\frac{\mu\text{mol}}{\text{m}^2\,\text{s}}$, the latter value corresponding to measured light conditions at the sea floor on the Peregrino field. The flow rates ($\mathcal{F}$) used were 0.04, 0.07, or 0.09 $\frac{\text{m}}{\text{s}}$. The amounts of sediment ($\mathcal{S}$), applied at the beginning of the experiment (T0), were 600, 900, or 1200 g per chamber. The chosen amounts resulted in a sediment coverage (SC) of the CA ranging from uncovered to completely covered, implying that also non- and very low sediment covered CA are included in the analysis. Six parallels (i.e. biological replicates) of rhodolites covered with healthy CA were placed in each chamber, so each of the 21 experiments was conducted with six samples.

Laboratory imaging   In addition to the measurements reported in [Figueiredo et al., 2015], digital images of the CA samples were taken to investigate responses of the samples in form of color and size changes during the weekly examination. For image recording, the CA rhodolites were removed from their chambers and manually placed in a transparent, water-filled cylinder together with a color reference plate (Figure 7.2 and Figure 7.3 A). The reference plate, essential for a comparative study of the color over time, can be seen in the upper part of the images (Figure 7.3 A upper left). The images were taken with a Canon PowerShot G10, which was placed on a tripod to control the distance (15 cm) between camera and samples. The camera was set to automatic mode with macro function and internal flash on. To eliminate light refractions and reflections, samples and camera, the latter protected by a waterproof housing, were placed underwater inside the cylinder (Figure 7.2). The digital images $\{\mathbf{I}_n\}_{n=1,...,630}$ had a resolution of 3456×2592 pixels (w $= 3456$, h $= 2592$). Each week 63 images (21 chambers $\times$ 3 images, 2 samples per image) were taken summing up to $N = 630$ images for the whole experiment (T0,...,T9).

7.1.2   Methods

The basic idea for extracting visual CA features from the recorded images is to segment the live CA in each image $\mathbf{I}_n$ by classifying each pixel in each image $\mathbf{I}_n$ to be either live/stressed CA or other, including dead CA. A pixel classifier is trained using data collected from the annotations provided by the experts (see below). This pixel classifier is build by applying unsupervised

Table 7.1: The Chamber configurations for the CCF design. The configurations for the 21 chambers for light intensity ($\mathcal{L}$), flow rate ($\mathcal{F}$), and sediment amount ($\mathcal{S}$) are displayed. Replicated configurations are grouped.

Chamber	$\mathcal{F}\left[\frac{\mathrm{m}}{\mathrm{s}}\right]$	$\mathcal{L}\left[\frac{\mathrm{\mu mol}}{\mathrm{m}^2 s}\right]$	$\mathcal{S}\,[\mathrm{g}]$
1	0.04	3.0	600
2	0.04	3.0	1200
3 4 5	0.04	6.6	900
6	0.04	10.0	600
7	0.04	10.0	1200
8 9	0.07	3.0	900
10	0.07	6.6	600
11 12	0.07	6.6	900
13	0.07	6.6	1200
14	0.07	10.0	900
15	0.09	3.0	600
16	0.09	3.0	1200
17 18 19	0.09	6.6	900
21	0.09	10.0	600
21	0.09	10.0	1200

Figure 7.2: The camera setup for imaging the samples. A schematic layout of the underwater camera setup is presented, consisting of a water filled cylinder containing the two samples, the color reference plate and the camera in the waterproof housing fixed on a tripod. The reference plate is consisting of 6 color circles and a 11 gray scales ranging from black to white. Colors are referring to the RGB triplets (left - right): (0,0,255), (255,0,0), (255,0,255), (0,255,0), (0,255,255), (255,255,0).

learning (Section 2.7.1) in form of clustering of the image feature space. After applying the clustering, the derived prototypes are either assigned to the positive class representing live CA, including stressed CA, or to the negative class representing the background, dead CA and bare substratum. After this, BMU matching is applied to classify all image pixel. From the pixel-wise classification, the regions of live/stressed CA are determined for each imaged sample at each time point. Finally, color and size information of segmented CA samples are extracted. The individual steps for image processing, pixel classification and data extraction are described in detail in the following.

Manual annotation   The manual annotation of the CA samples was carried out using BIIGLE v. 1.20 (Section 3.3). Biological experts from the Instituto Biodiversidade Marinha were manually placing annotations on the images using the BIIGLE software.

Here, an annotation again exists of a region $\mathbf{r}_n^{(j)}$ in an image $\mathbf{I}_n$, a class label $\omega_n^{(j)}$ and a biological expert also called observer o:

$$a_n^{(j)} = (\mathbf{r}_n^{(j)}, \omega_n^{(j)}, o).$$

For the annotation, the observers have to select a class $\omega \in \{$"live CA","stressed CA", "dead CA", "bare substratum"$\}$ and place single point regions or customizable rectangle regions (Figure 7.4) on the images.
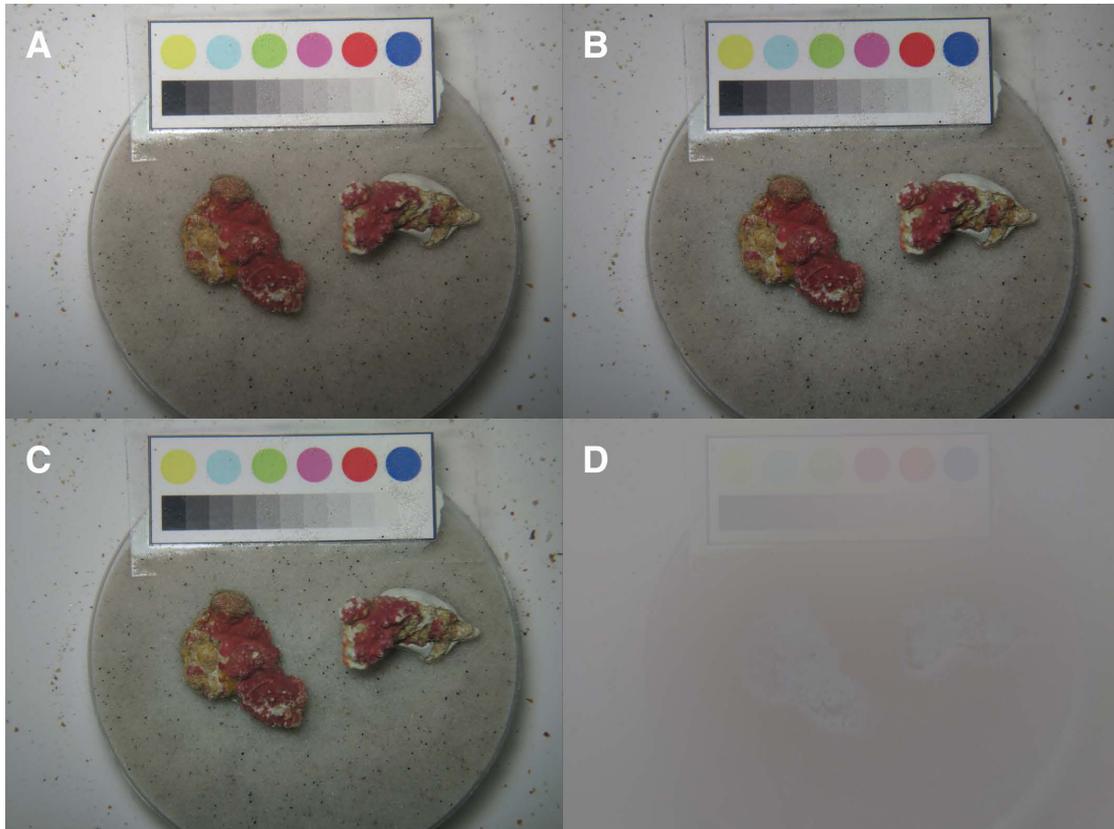
Figure 7.3: The preprocessing for the laboratory images. The original image (A) is first illumination corrected (B) and then color corrected (C). In order to illustrate the effect of the preprocessing a difference image $\frac{\mathbf{I}_n - \check{\mathbf{I}}_n}{2} + 128$ is computed (D). The color reference plate is located in the upper part of the images and provides the input for the preprocessing. The two CA samples are located below the reference plate. Comparing the result image from the illumination correction (B) with the original image (A), a slight decrease in brightness can be noticed. Due to the stable illumination conditions during image acquisition only little differences between the images can be recognized visually. The color correction (C) is changing the colors very little as only minor color shifts appear. Most of the visible differences in the difference image (D) are caused by the illumination correction.

These annotations provide examples for the later ML training step. The annotation strategy is chosen to speed up the manual annotation process and simultaneously cover the whole variation between and within each annotation class over time solving the AC challenge. During the manual annotation process it could be observed that a differentiation between the annotation classes live CA and stressed CA is problematic regarding the reproducibility, although no study of observer agreement was conducted. Anyway, a differentiation is unnecessary for the purpose of investigating the color change of the CA samples, as the color of the CA is expected to change with the level of stress. The classes for the ML are therefore reduced to a positive and a negative class ($\hat{\omega} \in \{0, 1\}$). The $a_n^{(j)}$ with $\omega_n^{(j)} \in \{\text{"live CA", "stressed CA"}\}$ represent the so called positive class $\hat{\omega} = 1$ in the segmentation and the $a_n^{(j)}$ with $\omega_n^{(j)} \in \{\text{"dead CA", "bare substratum"}\}$ represent the so called negative class $\hat{\omega} = 0$.

All images and annotations can be accessed via BIIGLE (`https://ani.cebitec.uni-bielefeld.de/biigle/bin/`). To browse the data, a login with username `pemca` and password `pemca` is required. The data are stored in the project `Pemca3rd`, with the transects $T0, ..., T9$. Image material is also published under DOI: 10.4119/unibi/2775316.

Image Preprocessing Before the feature extraction and the ML can be applied, the recorded images need to be preprocessed. Although images were recorded in an optimized imaging setup, variation in illumination can be observed. Colors are affected by the surprisingly varying numbers of particles in the water (DOP challenge) and possible due to influences of varying natural illumination. The latter might be caused by varying natural illumination in the lab. Images are therefore illumination and color corrected in the preprocessing ($\mathbf{I}_n \mapsto \check{\mathbf{I}}_n$).

Illumination fluctuations in the images are compensated by applying the local space average color scaling (LSACS) algorithm (Section 5.2.6) on each image individually. The LSACS image $\check{\mathbf{I}}_n$ is computed using a Gaussian blurring with $\sigma = 0.093 \cdot \max(\mathrm{w}, \mathrm{h})$. As the images were captured in sRGB [Stokes et al., 1996], which assumes a gamma correction of 2.2, the images are gamma corrected and then divided pixel- and channel-wise (c) by the corresponding Gaussian filtered image $\mathbf{G}_n$:

$$\check{\mathbf{I}}_n(x, y, c) = \left(\frac{(\mathbf{I}_n(x, y, c))^{2.2}}{2 \cdot \mathbf{G}_n(x, y, c)} \right)^{1/2.2}.$$

An example for an illumination corrected image is displayed in Figure 7.3 (upper right).

For the color correction the color reference circles are used. The color circles on the reference plate are detected in each image using a Hough circle transformation, [Yuen et al., 1990]. The Hough transform computes a list of circle candidates and those that show a low color variance inside are selected taking the spatial layout of the color plate into account, as well. The identified six color circles $\mathbf{r}_n^{(l)}$ ($l = 1, ..., 6$) of each $\check{\mathbf{I}}_n$ are then used to calculate one
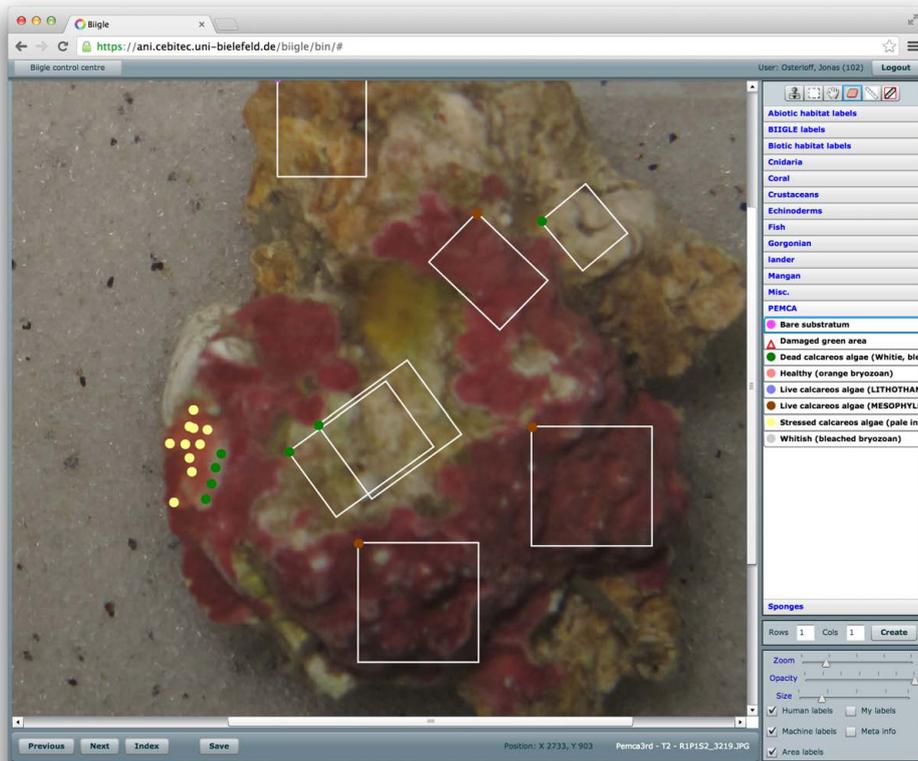
Figure 7.4: A screen-shot of the BIIGLE 1.20 system. Images of calcareous algae rodoliths were examined and annotated using the BIIGLE system. The biological experts, i.e. observers, were allowed to select annotations with single point regions or customizable rectangle regions. The single point annotations are represented as filled colored circles, the customizable rectangle annotations are outlined in white with a filled colored circle in the upper left corner of the individual rectangle. The colors of the filled circles indicate the class $\omega^{(j)}$ of the individual annotation $a^{(j)}$. Brown represents live CA, yellow stressed CA, green dead CA and Pink bare substratum.

RGB color triplet $\mathbf{c}_n^{(l)}$ for each circle over all its pixel values in $\mathbf{r}_n^{(l)}$.

$$\mathbf{c}_n^{(l)} = \begin{pmatrix} \underset{(x,y)\in\mathbf{r}_n^{(l)}}{\text{median}} \left(\check{\mathbf{I}}_n(x,y,r)\right) \\ \underset{(x,y)\in\mathbf{r}_n^{(l)}}{\text{median}} \left(\check{\mathbf{I}}_n(x,y,g)\right) \\ \underset{(x,y)\in\mathbf{r}_n^{(l)}}{\text{median}} \left(\check{\mathbf{I}}_n(x,y,b)\right) \end{pmatrix} \quad (7.1)$$

In addition, overall median color references are computed for each of the six color circles of all images (Figure 7.5).

$$\bar{\mathbf{c}}^{(l)} = \begin{pmatrix} \underset{n}{\text{median}} \left(\mathbf{c}_n^{(l)}(r)\right) \\ \underset{n}{\text{median}} \left(\mathbf{c}_n^{(l)}(g)\right) \\ \underset{n}{\text{median}} \left(\mathbf{c}_n^{(l)}(b)\right) \end{pmatrix} \quad (7.2)$$

For each image $\check{\mathbf{I}}_n$ and color channel $c \in \{r,g,b\}$, the individual, channel-wise exponent for a gamma correction is computed using the color RGB triplet $\mathbf{c}_n^{(l)}$ (Equation 7.1) of the individual image and the overall median color references (Equation 7.2).

$$\gamma_n(c) = \underset{l}{\text{median}} \left(\frac{\log\left(\frac{\mathbf{c}_n^{(l)}(c)}{255}\right)}{\log\left(\frac{\bar{\mathbf{c}}^{(l)}(c)}{255}\right)} \right)$$

This image specific γ_n is applied channel-wise to all the images resulting in a new set of color corrected images (Figure 7.3 lower left)

$$\check{\mathbf{I}}_n(x,y,c) = \left(\check{\mathbf{I}}_n(x,y,c)\right)^{\gamma_n(c)}.$$

A difference image ($\frac{\mathbf{I}_n - \check{\mathbf{I}}_n}{2} + 128$) is presented in Figure 7.3 (lower right) to demonstrate the impact of the described illumination and color correction.

The layout of the reference plate is also used to correct zoom variations, which occurred for a significant number of images. The average distance $\bar{d}$ between the neighboring color circles (e.g. green and magenta) on each plate is evaluated from the five Euclidian circle distances $d_{l,l+1}$

$$\bar{d} = \frac{1}{N} \sum_n \frac{1}{5} \bar{d}_n$$

with

$$\bar{d}_n = \sum_{l=1}^{5} d_n^{(l,l+1)}.$$

The ratio of the average distance and the observed individual distances in image $\check{\mathbf{I}}_n$ is used to normalize the extracted size feature (Paragraph: Calcareous algae feature extraction).
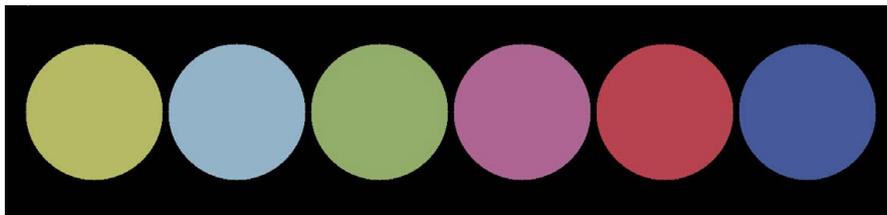
Figure 7.5: The average color reference plate. It is computed by averaging the detected reference plates from all images used in this study (see text for details). Colors are referring to the RGB triplets (left to right) (181,185,100), (147,179,200), (146,173,106), (173,101,147), (183,67,80), and (69,88,154).

Feature Extraction   One important step in clustering-based image segmentation is the selection of appropriate features and the collection of a training set. Here, median RGB values extracted on a 11×11 pixel neighborhood (η_{11}) for each color channel, are used as features. Although these feature are only a basic representation of the image information it seemed to be sufficient for the initial task of calcareous algae segmentation by pixel classification as they showed the best performance for the evaluated feature types (Paragraph: Parameter optimization).

$$\mathbf{v}_n^{(x,y)} = \begin{pmatrix} \underset{(x',y')\in\eta_{11}(x,y)}{\text{median}} \left(\breve{\mathbf{I}}_n(x',y',r) \right) \\ \underset{(x',y')\in\eta_{11}(x,y)}{\text{median}} \left(\breve{\mathbf{I}}_n(x',y',g) \right) \\ \underset{(x',y')\in\eta_{11}(x,y)}{\text{median}} \left(\breve{\mathbf{I}}_n(x',y',b) \right) \end{pmatrix}$$

In order to train and test a segmentation algorithm, class labels need to be assigned to the extracted feature vectors $\mathbf{v}_n^{(x,y)}$. As stated earlier, the annotation classes live CA and stressed CA form the positive class $\hat{\omega} = 1$. The $\{a_n^{(j)}\}$ are used to assigned a positive or negative class label to each feature vector $\mathbf{v}_n^{(x,y)}$:

$$\hat{\omega}_n^{(x,y)} = \begin{cases} 1 : & \exists j \;|(x,y) \in \mathbf{r}_n^{(j)} \wedge \hat{\omega}_n^{(j)} = 1 \\ 0 : & \text{else} \end{cases}$$

To reduce the computation time, features are extracted with a step width of $s = 2$ (Figure 2.3a) and are therefore in the following referred to as $\mathbf{v}^{(i)}$. The collected set of feature vectors $\{\mathbf{v}^{(i)}\}$ contains about 2.5 million elements.

Machine learning   In order to apply unsupervised learning, the H^{2}SOM algorithm [Ontrup et al., 2002] (see Section 2.7.1 Hierarchical Hyperbolic Self Organizing Map) is applied, as done to segment the cold water coral in 6.1.2. H^{2}SOMs are trained with different parameter settings, like the number of prototypes, on different feature types and sets. The best parameter-set was determined using a customized criterion described in the next paragraph.

The best H^{2}SOM is trained using $30 \times |\{\mathbf{v}_{(i)}\}|$ iterations, an exponentially decreasing learning rate of $\alpha = [0.99, 0.1]$, an eight-neighbor topology and three rings and is therefore consisting of 161 prototypes $\{\mathbf{u}^{(k)}\}_{k=1..161}$. $\{\mathbf{v}^{(i)}\}$ for training the final H^{2}SOM are selected by choosing 12.5% of all $\{\mathbf{v}^{(i)}\}$ image wise from a small subset (2%) of all images $\breve{\mathbf{I}}_n$.

The trained H^{2}SOM is used to assign a feature vector $\mathbf{v}^{(i)}$ to one of the H^{2}SOM prototypes $\{\mathbf{u}^{(k)}\}$ with ($k = 1..161$). This prototype is selected by the so called Best Matching Unit (BMU) criterion $\mu(\mathbf{v}^{(i)})$ (Equation 2.1) using the Euclidian distance.

Prototype class mapping In order to select prototypes representing the positive class, to tune the training parameters of the H^{2}SOM algorithm and for feature selection, the precision of each $\mathbf{u}^{(k)}$ is computed as follows. It is simulated that each $\mathbf{u}^{(k)}$ is representing the positive class. The $\{\mathbf{u}^{(k)}\}$ constitute a Voronoi tessellation of the feature space

$$V^{(k)} = \{\mathbf{v}^{(i)}|\ \mu(\mathbf{v}^{(i)}) = k\}$$

with $|V^{(k)}|$ the number of feature vectors that have been assigned to $\mathbf{u}^{(k)}$ and therefore are located in $V^{(k)}$. Each Voronoi cell $V^{(k)}$ is now analyzed for the number of feature vectors which stem from pixel positions annotated positive ($\hat{\omega}^{(i)} = 1$):

$$V^{(k,+)} = \{\mathbf{v}^{(i)} \in V^{(k)} \mid \hat{\omega}^{(i)} = 1\}$$

Thus, $|V^{(k,+)}|$ is the number of feature vectors in $V^{(k)}$ with a positive class label, i.e. the number of true positives (TP) and $|V^{(k)}|$ is the total number of positives in each $V^{(k)}$. Similar as defined in Equation 2.3, the relation

$$\mathsf{prec}^{(k)} = |V^{(k,+)}|/|V^{(k)}| = |TP|/(|TP| + |FP|) \tag{7.3}$$

is the precision of prototype $\mathbf{u}^{(k)}$. It quantifies how well a prototype $\mathbf{u}^{(k)}$ resembles the features of live CA. Since it is not expected that all $\mathbf{v}^{(i)}$ with $\hat{\omega}^{(i)} = 1$ are perfect representatives, only the prototypes $\mathbf{u}^{(k)}$ with highest $\mathsf{prec}^{(k)}$ will represent the positive class (Paragraph: Segmentation).

Parameter optimization $\mathsf{prec}^{(k)}$ (Equation 7.3) of each $\mathbf{u}^{(k)}$ is used to tune the training parameters of the H^{2}SOM and the feature types as stated earlier. Different feature types and different training settings are tested to find the best parameters and feature types. Median RGB values and histogram features (see Section 2.3.2) with different binnings (4, 8, or 16 bins per color channel) are tested with different pixel neighborhoods $\eta_5, \eta_9, \eta_{11}$. Also different sizes of random selections (1%, 2%, 5%, 10%) of images and amounts of feature vectors $\mathbf{v}^{(i)}$ (50%, 12.5%, 6.25%) per image are tested as well. $\mathsf{prec}^{(k)}(h)$ is computed for each training set h. The index $\{k'\}_1^{161}$ sorts all $\{\mathsf{prec}^{(k)}\}$ so that $\mathsf{prec}^{(k')} > \mathsf{prec}^{(k'+1)} \forall k'$. A score λ for each parameter-set h is computed using the sorted $\mathsf{prec}^{(k')}(h)$:

$$\lambda(h) = \sum_{k'=1,...,10} \mathsf{prec}^{(k')}(h)$$

The parameter-set with highest λ is then chosen for the final pixel wise segmentation.

Segmentation   In order to segment CA, a pixel classifier needs to be build from the trained H^{2}SOM. $\mathbf{u}^{(k)}$ need to be selected representing the positive class. This step is automated using the sorted $\text{prec}^{(k')}$ of each $\mathbf{u}^{(k')}$ in the following way. The first ϵ prototypes $u^{(k')}$ are chosen to represent the positive class, they are the prototypes for typical live CA color features. ϵ is chosen so at least 80% of the entire positive training data (i.e. $\hat{\omega}^{(i)} = 1$) are covered. Here, the set of the first $\epsilon = 16$ "most live calcareous algae-like" prototypes is modeling the positive class. The classification function is then defined as:

$$\Delta(\mathbf{v}^{(i)}) = \begin{cases} 1: & \mu'(\mathbf{v}^{(i)}) <= \epsilon \\ 0: & \text{else} \end{cases}$$

with

$$\mu' = \underset{k'}{\arg\min}\{||\mathbf{u}^{(k')} - \mathbf{v}^{(i)}||\}.$$

Finally the images are segmented on a pixel level by applying $\Delta(\mathbf{v}_n^{(x,y)})$, where pixel position (x, y) are marked as positive and therefore live CA where $\Delta(\mathbf{v}_n^{(x,y)}) = 1$ and as background where $\Delta(\mathbf{v}_n^{(x,y)}) = 0$.

The determined binary images are then post-processed with an opening filter mask [Gonzalez et al., 2008] of a size 10×10 pixels to remove small areas of false positive classifications. The post-processed binary images $\{\mathbf{B}_n\}$ represent the final segmentation mask.

Evaluation   The segmentation results are evaluated by comparing the manual annotations $a_n^{(j)}$ with the segmented areas in $\mathbf{B}_n$ by computing recall, precision and false positive rate (FPR). The accuracy is not considered due to the annotation strategy, annotating only examples for the annotation classes, so no precise outline of CA is available for the images.

The number of pixel positions annotated and classified positive are the true positives (TP):

$$\text{TP} = |\{(x, y)|\ \hat{\omega}_n^{(x,y)} = 1 \wedge \mathbf{B}_n(x, y) = 1\}|,$$

the number of pixel positions annotated negative and classified positive are the false negatives (FN):

$$\text{FP} = |\{(x, y)|\ \hat{\omega}_n^{(x,y)} = 0 \wedge \mathbf{B}_n(x, y) = 1\}|,$$

and the number of pixel positions annotated positive and classified negative are the false positives (FP):

$$\text{FN} = |\{(x, y)|\ \hat{\omega}_n^{(x,y)} = 1 \wedge \mathbf{B}_n(x, y) = 0\}|,$$

and the number of pixel positions annotated negative and classified negative are the true negatives (TN)

$$\text{TN} = |\{(x, y)|\ \hat{\omega}_n^{(x,y)} = 0 \wedge \mathbf{B}_n(x, y) = 0\}|.$$

Recall, precision and FPR are then computed as defined in Section 2.10 (Equation 2.4, 2.3, 2.5).

Calcareous algae feature extraction In order to analyze the possible biological impact of sedimentation on calcareous algae, the size and the average color of the segmented areas in the images are computed. The size (A_n) of the live calcareous algae area is measured as the amount of pixels marked with 1 in each $\mathbf{B}_n$. It is weighted by the ratio of the average color circle distance of circle centers $\bar{d}$ and the average circle center distance $\bar{d}_n$ of the color reference plate from the corresponding $\breve{\mathbf{I}}_n$.

$$A_n = \frac{\bar{d}}{\bar{d}_n} \sum_{x,y} \mathbf{B}_n(x,y)$$

The relative size $\hat{A}_n = \frac{A_n}{A_{n_0}}$, with A_{n_0} being the live CA size at the first measurement of the same imaged samples, is used as one variable. The HSV color space (Section 2.2.3) is used for measuring the color of the segmented CA. It is computed after computing the mean color $\bar{\mathbf{c}}_n^{(\mathrm{CA})}$ of the segmented CA in each image $\breve{\mathbf{I}}_n$ with

$$\bar{\mathbf{c}}_n^{(\mathrm{CA})} = \frac{1}{\sum_{x,y} \mathbf{B}_n(x,y)} \sum_{x,y} \mathbf{B}_n(x,y) \cdot \breve{\mathbf{I}}_n(x,y).$$

7.1.3 Results

The manual annotation resulted in 2404 annotations for live CA, 2834 annotations for stressed CA, 4358 annotations for dead CA and 43 annotation for bare substratum on all $\{\mathbf{I}_n\}_{n=1,\ldots,N=630}$. The number of pixels per annotation class looks a bit different: ca. 24×10^6 pixels for live CA, ca. 10×10^6 pixels for stressed CA, ca. 10×10^6 pixels for dead CA and ca. 0.5×10^6 pixels for bare substratum. Most of the live CA annotations ($\approx 90\%$) and bare substratum annotations ($\approx 98\%$) consist of rectangle shaped regions, while for the classes stressed CA the dead CA more than 60% of the annotations consist of point regions.

In 15% of the images used in the analysis, the reference color circles could not be detected automatically. In these images the circles were marked manually or only a subset of detected circles was used for the color correction. Although reflections were avoided on the CA samples, serious reflection problems were observed on the color reference plates in some of the images. The images from T0 and T1 had to be excluded from further analysis because of this problem.

Fluctuations in sample illumination are a well-known problem in computational image analysis and the majority of all images suffered at least partially from it. The described preprocessing compensated this effect (Figure 7.3 (B)). Visual inspection showed that the channel-wise gamma correction produced robust results.

From the segmentation results a recall value of 0.89 and a precision of 0.03 is estimated. The FPR was estimated for the negative class and also for the annotation classes dead CA and bare substratum and are presented in Table 7.2. A visualization of selected segmentation results are also displayed in Figure 7.6.

Table 7.2: The results for the evaluation of the segmentation. The very low values for precision were expected, since a realistic quantification of the FP is not possible in the dataset (see text for details). The low FPR though indicates that the segmentation generates low FP in the regions annotated as dead CA or bare substratum.

Recall	Prec.	FPR		
		negative class	dead CA	bare substratum
0.89	0.03	0.02	0.02	0.05
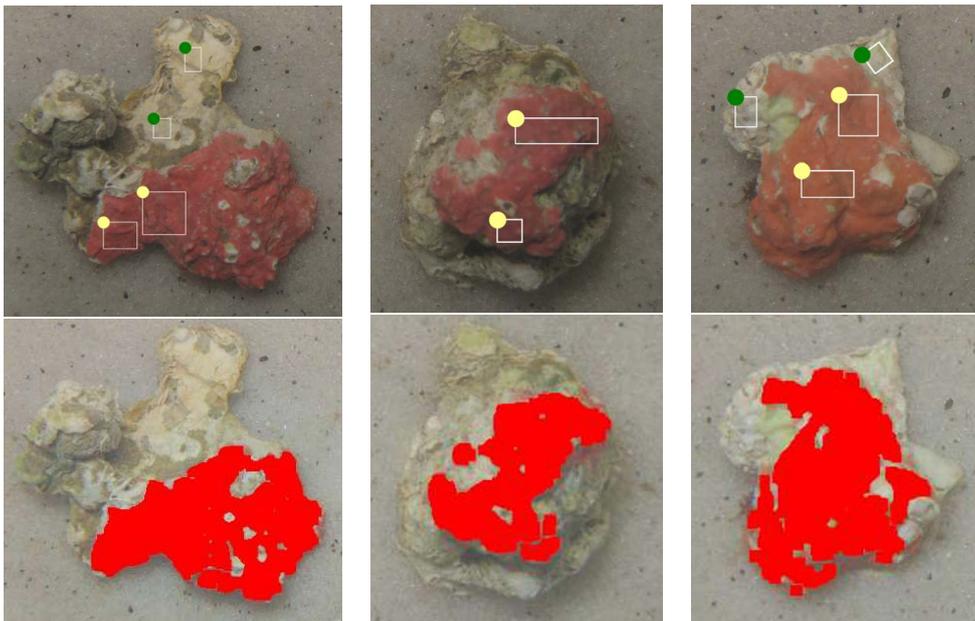
Figure 7.6: Examples of manually labeled rhodolites (left) and the corresponding segmented rhodolites (right). White rectangles with a green dot mark regions annotated as dead CA and rectangles with a yellow dot mark regions annotated as stressed CA. The pixels classified to be live CA are highlighted in red.

Calcareous algae features were extracted from remaining $N' = 504$ images. Changes of size and color can be noticed for different configurations. Different treatment parameters seem to produce different responses of the CA samples (see Figure 7.7 for an example).

Averages of the CA features were computed for each week (T0,...,T9) over all chambers to investigate the temporal changes of the CA features independent of the other predictors. Looking at the average color features $\tilde{\mathbf{c}}_{\mathcal{T}}^{(\mathrm{CA})}$ in the HSV color space (see Section 2.2.3 for details) it is noticeable that hue and saturation do change over time while the value is rather constant (see Figure 7.8). The averages extracted sizes $\tilde{A}_{\mathcal{T}}$ show a clear decrease over time (Figure 7.8).

7.1.4 Discussion

The selected strategy for the manual annotation in this study was to apply an example based annotation approach, rather than to fully outline the different CA regions. The rationale behind the chosen strategy was that color changes were expected to be time and/or treatment related and that the use of a such an annotation approach (rectangles and points) would balance the time spent on annotation and the amount of training data needed.

The results of the preprocessing steps show that the illumination fluctuations could be reduced significantly. Furthermore, the reference plate enabled to achieve color constancy and to correct camera zooming for all images.

Due to the chosen annotation strategy, no representative ground truth for the negative class, i.e. pixel regions that show no CA, exists. In principle, the negative class could have been constructed by a random selection from the non-annotated regions of the images, as done in Section 6.1.3. However this random selection might have introduced some false-negatives to the negative class. This motivated to apply an unsupervised learning approach. The prototype label mapping for the trained H^{2}SOM followed a conservative strategy to classify only 80% of the positive training data as live/stressed CA. Despite of this strategy, the segmentation results show a high recall of 0.89. Based on the selected example based manual annotation strategy the precision is low, as expected. A posterior visual inspection of the results showed that the selected approach produced many false positives that actually turned out to be true positives, as expected. Computing the precision is therefore not a reasonable assessment for the accuracy of this segmentation system (Figure 7.6). The FPR scores is the better evaluation score here. It is low, indicating that almost none of the pixels annotated as dead CA or bare substratum were segmented into live or stressed CA regions. The FPR for the bare substratum is noticeable, but is interpreted as still low.

The extracted CA features indicate some possible correlations with the investigated parameters including time. A detailed analysis of predictor and response variables is presented in Section 8.2.
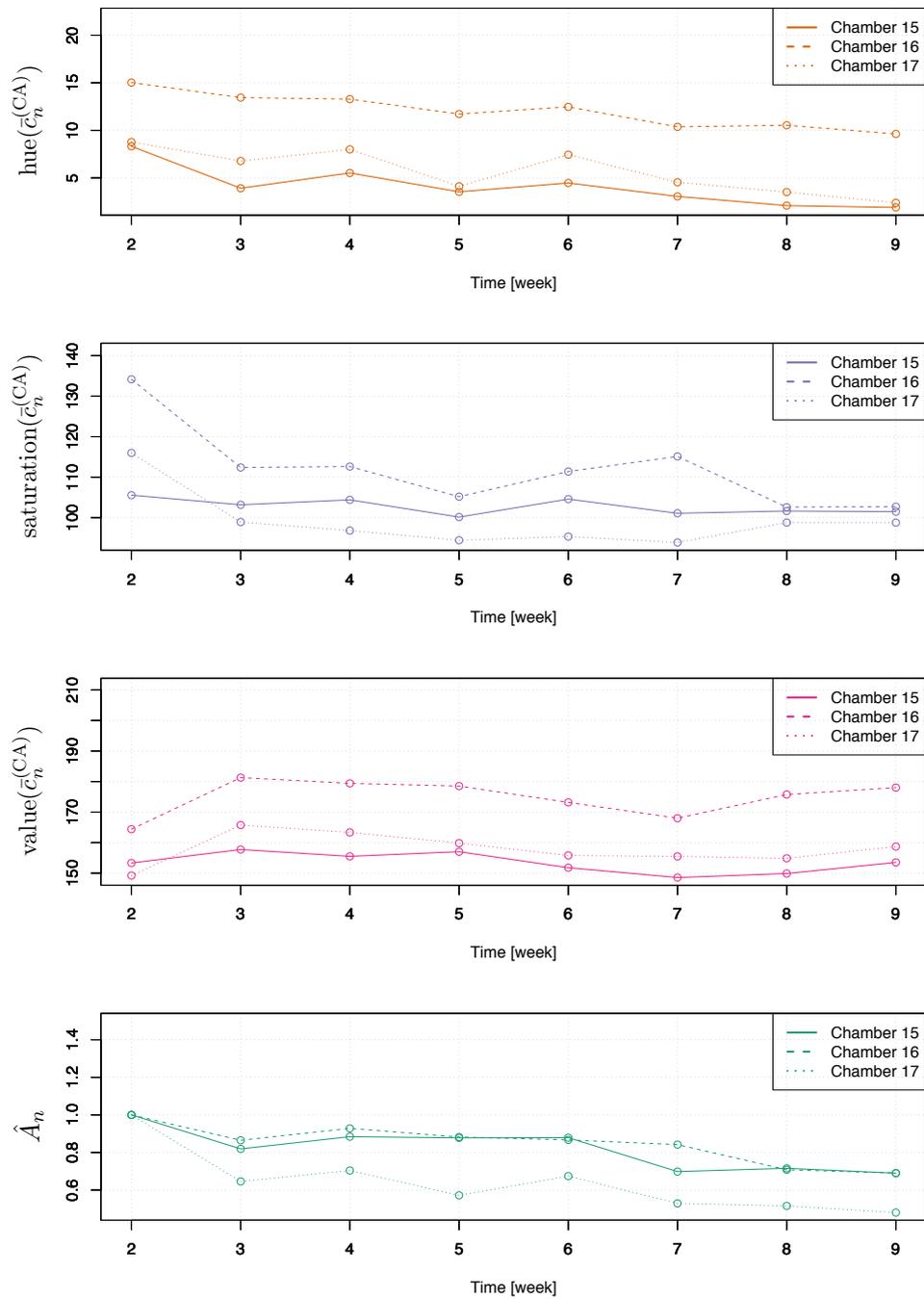
Figure 7.7: The extracted calcareous algae features for chambers 15-17. Color features are presented in the HSV color space (see Section 2.2.3 for details). All chamber feature different $\mathcal{S}$ concentration at highest $\mathcal{F}$ (see Table 7.1 for details).
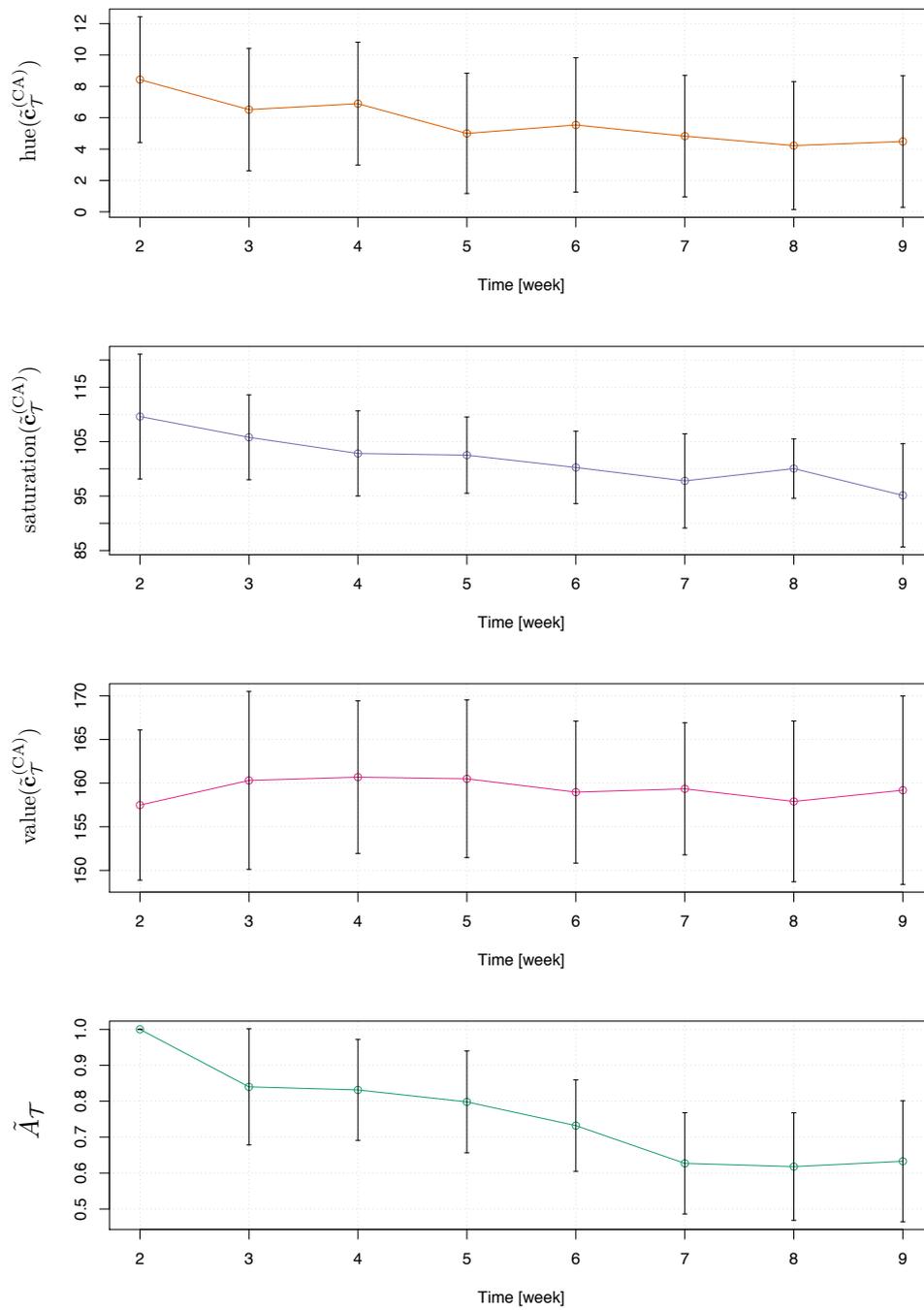
Figure 7.8: The weekly average calcareous algae features: relative size $\tilde{A}_{\mathcal{T}}$ and CA color $\tilde{\mathbf{c}}_{\mathcal{T}}^{(\mathrm{CA})}$. Color features are presented in the HSV color space (see Section 2.2.3 for details). Error bars are showing the standard deviation.

7.1.5   Conclusion

The presented CV system allows to quantify changes in size and color of CA sample from digital images. In contrast to *in-situ* monitoring, imaging conditions can be optimized in a *wet-lab* experimental setup. Nevertheless, a preprocessing of the images is most probably necessary to achieve a color constancy over time and therefore over all images, as done here.

Although the presented preprocessing pipeline allows to enhance the image quality and to compensate feature shifts caused by the *wet-lab* experimental setup, some of the experimental steps can still be improved. For instance, in most of the excluded images, the color reference plate showed strong reflections due to a suboptimal material surface and the angle between camera and reference plate. The strong reflections may be avoided by selecting a material that gives minimal reflections and by optimizing the imaging setup using an optimized stable artificial light source only. Reference plate and sample should be illuminated evenly throughout the whole image. The optimized light setup might also reduce the need for the preprocessing. Additionally, using a camera with manual settings enables the use of a fixed white balance that reduces possible fluctuations in color as done for LoVe image set 2. In order to increase the contrast between OOI and background a different colored background for imaging should be chosen (e.g. blue / green screen). This would reduce the possibility to produced FP on the background of the images. Furthermore, rotations of rhodolites around an axis not perpendicular to the image plane between the different time points of image acquisition (see Figure 8.10) could be observed, affecting mostly the extracted size $\hat{A}_n$. The use of plaster imprints for imaging the samples in combination with the background modifications would improve the reliability of the presented system. By experience even in laboratory studies these conditions are often hard to achieve throughout the whole experiment. The presented semi-automated software solution can be used to evaluate the images even if not recorded under the optimized conditions, as shown here.

The effect of the AQ challenge is noticeable as a low reproducibility for assigning either the class label "live CA" or "stressed CA" to identical image regions. This challenge though could be resolved in a straightforward as it turned out that the differentiation is unnecessary here. Extracting the average color for the combined classes enabled to quantify the stress levels as shown in Section 8.2. The AC challenge is solved by applying an unsupervised ML algorithm and also with the chosen example based annotation strategy. In the presented workflow the AU challenge was resolved by applying unsupervised ML. The strategy was not even to try to generate training data covering the whole live/stressed CA in each image. This affected the evaluation, which is therefore reduced to compute the precision and FPR only. The computed values for both scores and visual inspections of the segmentation result show that the CV system is reliable.

The presented methodological approach shows the clear potential to be applied to other *wet-lab* studies with other sessile species where color change is a stress indicator, e.g. tropical corals [Siebeck et al., 2006]. However, as individual species have species specific features (color, structure, shape, etc.) and their surroundings differ, species specific adjustment for computational analysis may be required.

8 Data science for environmental monitoring

In the previous chapters (6 and 7) species-specific information was extracted from digital images using CV. In order to start to understand the species behavior the analysis can not end and this point. Species behavior needs to be related to other variables to unveil functional relations, generating new biologically relevant information. "Data science is the study of the generalizable extraction of knowledge from data" [Dhar, 2013]. This data are heterogeneous and unstructured for instance encapsulated in images and require the integrations and interpretations by applying for instance computer science methods [Dhar, 2013]. The whole process of information extraction from images and the integration from other sensor data can be summarized under the term data science. As the information extraction from images was already discussed in the previous chapter, the focus of this chapter lies on the integration of multiple sensor data and its analysis for functional relations.

Two studies are presented here. First, exemplary for *in-situ* studies, *Lophelia pertusa* behavior extracted from images (Section 6.1) is integrated with additional sensor data, also recorded by the image recording FUO LoVe and functional relations among the integrated sensors are investigated. Second, behavior responses in a *wet-lab* study of the investigated species *Mesophyllum engelhartii* also extracted from digital images (Chapter 7.1) is analyzed for functional relations with the treatment variables of the experiment. Furthermore, the functional relations between the photosynthetic efficiency P measured during the experiment and the computed calcareous algae color $\bar{\mathbf{c}}_n^{(\text{CA})}$ is investigated. Both studies have already been published. Section 8.1 is based on publication 8 and Chapter 7.1 is based on publication 3.

8.1 In-situ: Lophelia pertusa

As stated in Section 6.1, monitoring the reef building cold water coral *Lophelia pertusa* is of high interest. In order to start to understand the relation and interaction of environmental parameters and *Lophelia pertusa* responses data science is needed. In the following, an initial approach is presented to identify possible correlations and giving insights to the complex marine environment. This is the first study, which investigates the functional relations of environmental sensor data, like temperature or salinity and species behavior in form of polyp activity (PA) for such a long period. In [Johanson et al., 2017] functional relations between environmental sensor data and polyp behavior of the cold water coral *Paragorgia aborea* were investigated. The temporal duration ($\approx$ 11 days) was much shorter, due to a manual estimation of the corals PA.

8.1.1 Material

In Section 6.1 visual features were extracted from images recorded at the LoVe observatory (LoVe image set 2). These features are polyp activity (PA) θ_t and coral color $\widetilde{c}_t^{(\text{coral})}$. For the

latter only the a*-channel is considered, as it quantifies the color change of the color reef, shown in Section 6.1:

$$\xi_t = \tilde{c}_{t,a*}^{(\text{coral})}.$$

The LoVe observatory consists of the satellite unit and an x-frame unit which is connected to the mainland. The x-frame is equipped with two acoustic Doppler current profilers (ADCP) and an echo sounder. The ADCPs are used to monitor the current velocity (speed) and direction as they are measuring the velocities in north, east and upward direction. The first ADCP is a long range (Frequency: 193.5 kHz) instrument with a bin size of 5 m and 2 m $-$ 262 m range. The second is a short range (Frequency: 600 kHz) ADCP with a bin size of 2 m and 0.5 m $-$ 40.5 m range. The satellite unit is equipped with the camera and flash and Chlorophyll, turbidity, pressure, hydrophone, temperature and conductivity sensors (Table 8.1). Furthermore, it is equipped with an eco-triplet for measuring chlorophyll, colored dissolved organic matter (cDOM) and turbidity.

For quality assurance of the different sensors, redundant sensors were tested for linear correlation and non redundant sensors were investigated for unrealistic measurements and reaches of sensor range limits. This resulted in neglecting several sensors for the later analysis. The eco-triplet is neglected as it showed unexpected variation in sampling intervals resulting in high variation of signals. Furthermore, sensor output signals of the triplet are most likely in a different order than documented, as for instance colored dissolved organic matter (cDOM) measurements are most likely Chlorophyll measurements. The turbidity sensor is excluded as the sensor was not calibrated for high turbidity, as measurements exceeded the range limit. Unrealistic readings were identified and neglected for ADCP bins encoding farther distances. Furthermore, due to missing recordings of the short range ADCP for the beginning of the analyzed time period only the long range ADCP is included in the following analysis. Other sensors not considered for the analysis are an echo sounder, a hydrophone and a compass.

The data recorded by the sensors together with the color features ξ_t and the PA θ_t values extracted from the image series form the final data collection that is subject to the statistical analysis. An overview for all data attributes and temporal resolutions is given in Table 8.1. To get a first impression, the raw sensor data f_t for each sensor is displayed in Figure 8.1. Time series of coral color and PA are presented in Figure 8.2. As can be seen in the plots the time series show gaps caused by problems with the sensors including the camera or break downs of the communication connection. Nevertheless, the raw time series of θ_t, ξ_t as well as temperature, conductivity and salinity show large scale trends and patterns all starting in May (Figure 8.1 and 8.2).

8.1.2 Methods

In order to generate new knowledge from the extracted image information, together with the environmental sensors, i.e. scalar sensors, from the LoVe observatory, several analyses were conducted. First, Pearson correlation is applied to unveil linear relations. Wavelet

Table 8.1: The LoVe observatory sensors. Details about manufacturer, measured variable and temporal resolution are included.

Sensor	Name	Measured variable	Symbol	Unit	resolution
ADCP long range	Nortek Continental	Current velocities in north, east, upward direction	$f_t^{(Vn_1)}$, $f_t^{(Ve_1)}$, ...	[m/s]	10 min
ADCP short range	Nortek Aquadop	Current velocities in north, east, upward direction	*	[m/s]	10 min
Chlorophyll	Seapoint Chlorophyll Fluorometer	Chlorophyll	$f_t^{(Chl)}$	[QSU]	60 min
Turbidity	AADI 4112A	Turbidity	*	[FTU]	60 min
Pressure	AADI 4117D	Depth Temperature	$f_t^{(Depth)}$ $f_t^{(Temp)}$	[m] [°C]	60 min
Conductivity	AADI 4319A	Conductivity Temperature Salinity	$f_t^{(Cond)}$ $f_t^{(TempC)}$ $f_t^{(Sal)}$	[S/m] [°C] [PSU]	60 min
ECO triplet	Wetlabs ECO Triplet-w	Chlorophyll cDOM NTU	*	[µg/l] [ppb] [NTU]	60 min

* not considered in the subsequent analysis (see text for details).
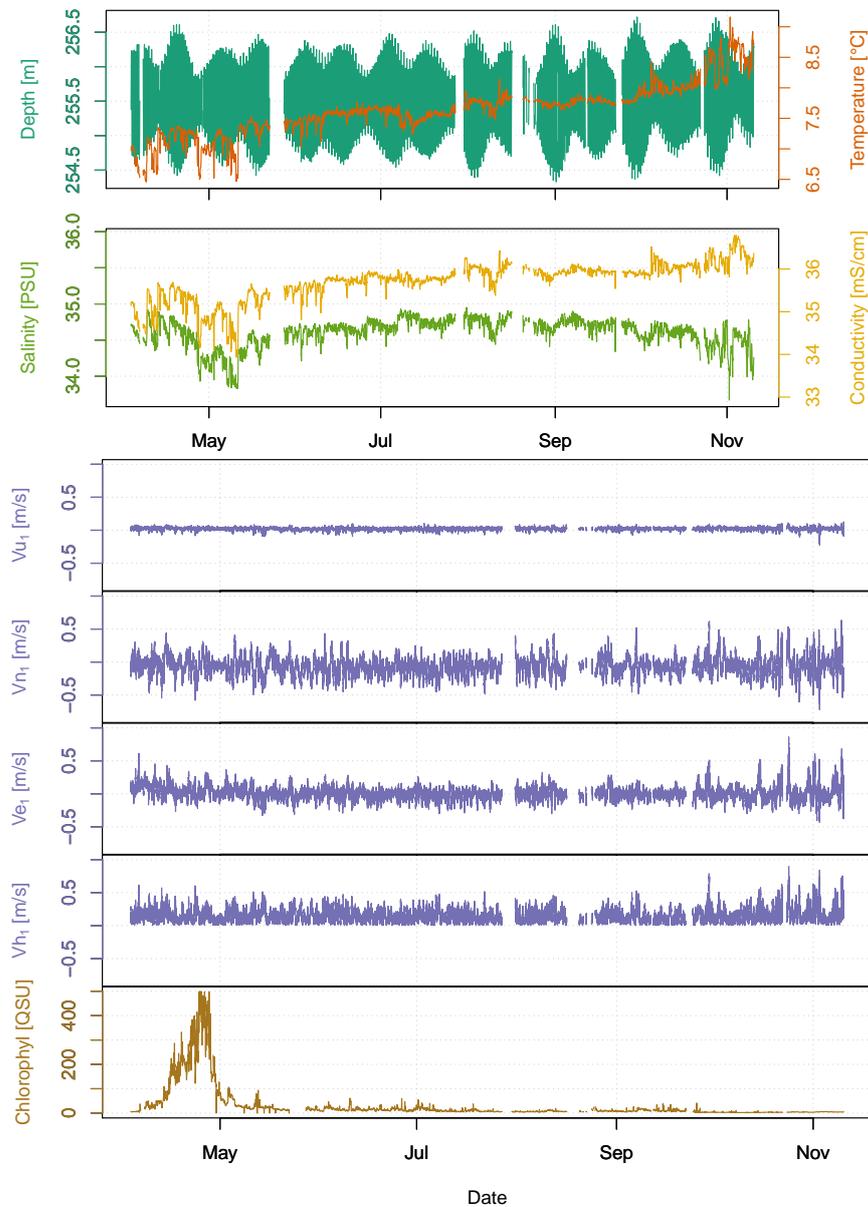
Figure 8.1: Plots for the measurements of selected LoVe sensors. Gaps in the series stem from sensor malfunctions. Starting from the top row, the first plot shows the measurements of the pressure sensor. This includes estimated depth (dark green) and temperature (orange). Next, the salinity (green) and conductivity (yellow) are presented. In the following four plots, the four read outs of the current velocity measurement closest to the ground are shown (purple). Current velocities were measured in upward direction (Vu_1), north direction (Vn_1) and east direction (Ve_1). Resulting horizontal current velocity (Vh_1) is computed from Vn_1 and Ve_1. In the last plot (bottom), the measurements from the Chlorophyll sensor are shown (brown).
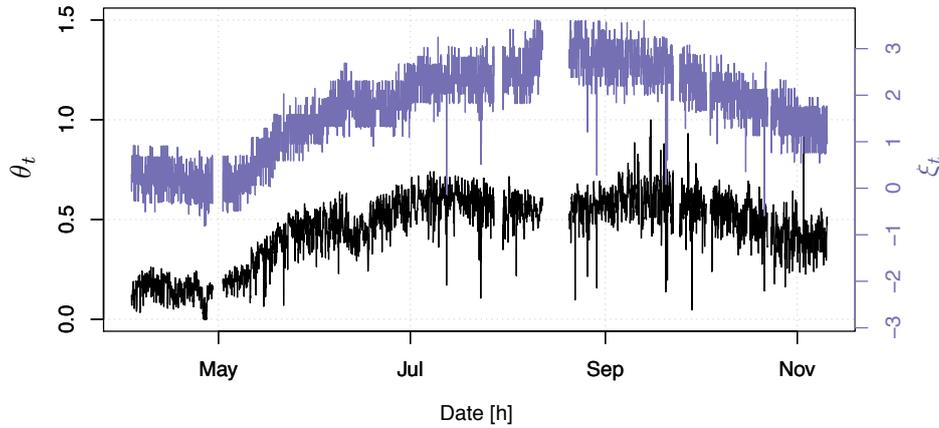
Figure 8.2: Coral color and polyp activity (PA) estimated from the LoVe images. The PA θ_t is colored in black and the coral color ξ_t (a*channel of the coral color $\tilde{c}_{t,a*}^{(coral)}$) is colored in purple. Gaps in the series stem from camera malfunctions.

transformation is then applied to identify periodicity in the data. Wavelet analysis is applied to investigate coherences of periodicity of the different time-series.

First, inter-correlation of the polyp behavior is investigated. The image region investigated for the PA (see Section 6.1) is divided into a grid sctructure with 12 evenly sized quadratic cells ($q = 1, ..., 12$) (Figure 8.3) containing $40,000$ pixels each. Polyp activity is then estimated for the 12 cells ($\{\theta_t^{(q)}\}_{q=1,...,12}$) and pairwise Pearson correlation is computed.

Second, possible correlation between extracted coral color ξ_t and θ_t is investigated. This includes the correlation analysis of daily averages $\bar{\xi}_t$ and $\bar{\theta}_t$ (see below). Again, Pearson correlation is applied computing the linear relationship between (θ_t, ξ_t) and $(\bar{\theta}_t, \bar{\xi}_t)$.

In order to unveil possible functional relations between PA θ_t and the other physical qualities measured by the scalar sensors f_t, like temperature $f_t^{(Temp)}$ or salinity $f_t^{(Sal)}$, several analyses are conducted. In contrast to the analyzes of inter-correlation and correlation of θ_t and ξ_t, it has to be considered that the different sensors of the LoVe observatory have different sampling rates and show multiple short episodes of sensor dysfunctions. Sensor dysfunctions additionally are causing different kinds of temporal delays between measurements of different sensors. Depending on the type of the analysis, this requires a mapping and/or interpolation for the different time series.

In order to test linear correlations between the image derived time series θ_t and the time series from scalar sensor data, for each scalar sensor f_t a time series ν_t, that has the same temporal resolution as the image time series, is computed. Given an original sensor time series f_t and the original time points $\{t | f_t \neq \emptyset\}$ of its measurements the new sensor time

119
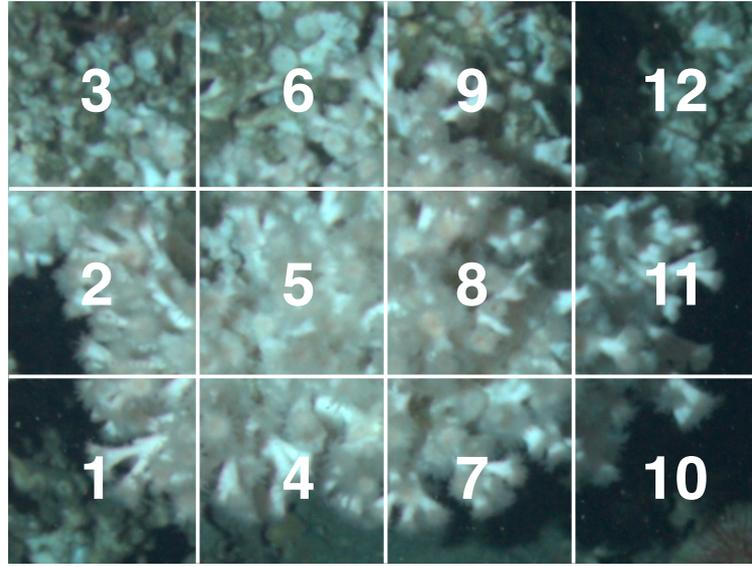
Figure 8.3: The area for the investigation of the polyp activity (PA) is divided into a grid structure with 12 cells ($q = 1, ..., 12$) to investigate for colonial coordinations. The number of polyps was not considered during the partitioning. The cells are equally sized with a width of 200 pixels and a height of 200 pixels.

series ν_t is computed by interpolation:

$$\nu_t = \frac{f_{t^{\geq}} - f_{t^{\leq}}}{t^{\geq} - t^{\leq}} \cdot (t - t^{\leq}) + f_{t^{\leq}}$$

with $t^{\leq} = \max\{t^*|t^* \leq t \wedge t^* \in \{t|f_t \neq \emptyset\}\}$ and $t^{\geq} = \min\{t^*|t^* \geq t \wedge t^* \in \{t|f_t \neq \emptyset\}\}$. Linear correlation is then tested pairwise between θ_t and ν_t (see Section 8.1.3). Furthermore, daily averages $(\bar{\theta}_t, \bar{\nu}_t)$ of θ_t and ν_t are computed and tested for correlation. For the latter, no linear interpolation is applied, ignoring days where no data were recorded either for θ_t and/or ν_t.

The second analysis aims at the identification of periodicity in single time series (e.g. θ_t or f_t) or in pairs of time series θ_t and f_t. It is conducted in the frequency domain and in the spatial domain via wavelet transformations using the R package "biwavelet" [Gouhier et al., 2016]. In contrast to a Fourier transformation based analysis, the wavelet transformation allows to identify periodicity in the data with limited temporal extent. Since the methodology behind the biwavelet package assumes equidistant time points for the analysis of a time series (and identical temporal resolution for a pairwise analysis) the time series have to be transformed accordingly. Estimated PA θ_t and coral color ξ_t with the original time points of imaging $\{t|\theta_t \neq \emptyset\}$ are therefore converted to have equidistant time points ($t - (t-1) = 1$ h) with a one hour distance by linear interpolation:

$$\theta'_t = \frac{\theta_{t^{\geq}} - \theta_{t^{\leq}}}{t^{\geq} - t^{\leq}} \cdot (t - t^{\leq}) + \theta_{t^{\leq}}$$

with $t^{\leq} = \max\{t^*|t^* \leq t \wedge t^* \in \{t|\theta_t \neq \emptyset\}\}$ and $t^{\geq} = \min\{t^*|t^* \geq t \wedge t^* \in \{t|\theta_t \neq \emptyset\}\}$. To allow for a pair wise analysis of image-derived time series with other sensor data f_t is mapped ($f_t \rightarrow \nu'_t$) to interpolate measurements for $\{t|\theta'_t \neq \emptyset\}$. The wavelet transformation is applied to each individual time series with a "morlet" mother wavelet function, followed by pairwise coherence wavelet analysis between θ'_t and all ν'_t.

8.1.3 Results

First, the inter-correlation of the PA of the different cells q was computed followed by the correlation analysis of θ_t and extracted coral color ξ_t. The results of the inter-correlation are presented in Table 8.2. All $\{\theta_t^{(q)}\}_{q=1,\dots,12}$ and θ_t correlate well except region $q = 10$. For (θ_t, ξ_t) a high Pearson correlation coefficient of 0.84 was computed and for $(\bar{\theta}_t, \bar{\xi}_t)$ an even higher correlation coefficient of 0.94 was computed (Table 8.3).

Next, the pairwise correlation analysis between θ_t and the scalar sensor data f_t were investigated. This was done with standard methods like exploratory data analysis and pearson correlation analysis. To plot PA for instance against temperature (Figure 8.4) linear interpolation was applied to temperature measurements ($f_t^{(\text{TempC})} \rightarrow \nu_t^{(\text{TempC})}$) to match the original image time points $\{t|\theta_t \neg \emptyset\}$ (see previous section). This matching also allows to compute the pearson correlation, which was computed to display linear relations of θ_t and other selected sensors (Table 8.3). Pearson correlation coefficients (corr) for pairwise correlation of θ_t and ξ_t are high (0.84) and for $\bar{\theta}_t$ and $\bar{\xi}_t$ even higher (0.94). Pairwise correlation between θ_i and $\nu_t^{(\text{Cond})}$, $\nu_t^{(\text{TempC})}$, $\nu_t^{(\text{Sal})}$ are not as high. θ_i and $\nu_t^{(\text{Chl})}$ are weak negatively correlated corr$(\theta_t, \nu_t^{(\text{Chl})}) = -0.56$ and corr$(\bar{\theta}_t, \bar{\nu}_t^{(\text{Chl})}) = -0.61$ respectively). Looking at the estimated coefficients for the current velocities it is noticeable that all are negatively correlated, but most of them are very low stating for no linear dependency between current velocity and PA. In general, estimated coefficients for the daily averages are higher.

The plot of the daily averages for temperature ($\bar{\nu}_t^{(\text{TempC})}$) against daily averages of PA ($\bar{\theta}_t$) visualize their correlation. For low temperatures, a low PA was estimated and with increasing $\bar{\nu}_t^{(\text{TempC})}$ also $\bar{\theta}_t$ increases until the maximum activity is reached. Finally $\bar{\theta}_t$ decreases again for $\bar{\nu}_t^{(\text{TempC})} > 8$ °C.

The periodicity of the different time series was analyzed using wavelet transformations. Linear interpolation was used to compute continuous time series with equidistant time points from θ_t and f_t (see above). The result of this analysis is visualized in Figure 8.5a. On the left, the significance of the periodicity for a given period length (here hours, y-axis) at a given time point (here hours, x-axis) is shown in a color map. A continuous horizontal deep red line at a value y' would indicate a periodicity of y' hours throughout the entire observation period. Although such a clear pattern cannot be seen, the PA shows some periods with a significant wavelet response for low periodicities which occur more frequently from September until mid November. Next to the color map, the global power plot shows the accumulated significance of the transformation response for each tested periodicity. Considering the entire period a

Table 8.2: The inter-correlation of polyp activity θ_t of the different cells $q = 1, ..., 12$ are computed as Pearson correlations. High coefficients were computed for all pairs of $(\theta_t^{(q)}, \theta_t^{(q')})$ with exception of region $q = 10$.

q	1	2	3	4	5	6	7	8	9	10	11	12	All
1	-	0.84	0.80	0.85	0.82	0.83	0.70	0.80	0.85	0.46	0.75	0.76	0.87
2	0.84	-	0.92	0.82	0.89	0.90	0.84	0.92	0.90	0.52	0.83	0.81	0.95
3	0.80	0.92	-	0.77	0.86	0.91	0.83	0.90	0.90	0.50	0.82	0.82	0.94
4	0.85	0.82	0.77	-	0.88	0.84	0.67	0.77	0.86	0.41	0.67	0.72	0.87
5	0.82	0.89	0.86	0.88	-	0.91	0.80	0.89	0.89	0.49	0.78	0.79	0.95
6	0.83	0.90	0.91	0.84	0.91	-	0.80	0.90	0.94	0.49	0.82	0.83	0.96
7	0.70	0.84	0.83	0.67	0.80	0.80	-	0.91	0.78	0.60	0.84	0.76	0.88
8	0.80	0.92	0.90	0.77	0.89	0.90	0.91	-	0.90	0.57	0.87	0.82	0.96
9	0.85	0.90	0.90	0.86	0.89	0.94	0.78	0.90	-	0.50	0.83	0.85	0.96
10	0.46	0.52	0.50	0.41	0.49	0.49	0.60	0.57	0.50	-	0.57	0.49	0.55
11	0.75	0.83	0.82	0.67	0.78	0.82	0.84	0.87	0.83	0.57	-	0.88	0.88
12	0.76	0.81	0.82	0.72	0.79	0.83	0.76	0.82	0.85	0.49	0.88	-	0.87
All	0.87	0.95	0.94	0.87	0.95	0.96	0.88	0.96	0.96	0.55	0.88	0.87	-

Table 8.3: Pearson correlation coefficients were computed pairwise for polyp activity (PA) and coral color (corr(θ_t, ξ_t)) and PA and selected scalar sensor data (corr(θ_t, ν_t)). Linear interpolation $f_t \rightarrow \nu_t$ is applied to compute correlations with the scalar sensor data. Coefficients for the daily averages $\bar{\theta}_t$ and $\bar{\xi}_t$ and $\bar{\nu}_t$ are given in the second column.

Sensor		corr(θ_t, ν_t)	corr($\bar{\theta}_t, \bar{\nu}_t$)
Coral color	ξ_t	0.84	0.94
Conductivity	$\nu_t^{(\text{Cond})}$	0.61	0.68
Temperature (conductivity sensor)	$\nu_t^{(\text{TempC})}$	0.48	0.54
Salinity	$\nu_t^{(\text{Sal})}$	0.48	0.54
Chlorophyll	$\nu_t^{(\text{Chl})}$	-0.56	-0.61
Current v. up (2 m $-$ 7 m)	$\nu_t^{(\text{Vu}_1)}$	-0.07	-0.21
Current v. north (2 m $-$ 7 m)	$\nu_t^{(\text{Vn}_1)}$	-0.06	-0.16
Current v. east (2 m $-$ 7 m)	$\nu_t^{(\text{Ve}_1)}$	-0.22	-0.37
Current v. horizontal (2 m $-$ 7 m)	$\nu_t^{(\text{Vh}_1)}$	-0.06	-0.13

24-hour a 24-hour periodicity is noticeable. The global power value is also high for longer periodicities including a periodicity close to the moon phase (approx. 700 h $\approx$ 29 days). Highest global periodicity can be reported for one of the longest analyzed periods (1412 h $\approx$ 59 days).

A stronger periodicity pattern of 24 hours can be observed for the current velocities (see Figure 8.5b). It also can be reported that current velocities in eastward direction have a local maximum at moon phase periodicity (not shown). Depth measurements show a clear tidal periodicity of 12.4 h. Conductivity, temperature and salinity all show no clear pattern over the whole time-period. Only some short periods with significant wavelet responses can be reported, but they show a very broad spectrum of periodicities.

Finally, a wavelet coherence is computed between θ_t' and $\nu_t'^{(\text{Vn1})}$, as both time series show a 24-hour pattern. The visualization of the wavelet coherence analysis (Figure 8.6) can be interpreted similar to the wavelet response visualizations (Figure 8.5a and Figure 8.5b). The color map encodes the coherence significance $[0, 1]$, a value of 1 would indicate a perfect coherence and a value of 0 would indicate no coherence at all. The global power plot shows the accumulated significance of the coherence for each periodicity. Several periods featuring a high 24-hour coherence are noticeable resulting in a local maximum close to 24 h for the global power. Highest global power is reached for a coherence periodicity of approx. 790 h $\approx$ 33 days.

8.1.4 Discussion

The results show that even simple linear regression can be applied to understand this complex environment. The inter-correlation of the PA of the q cells showed that all polyps behave in
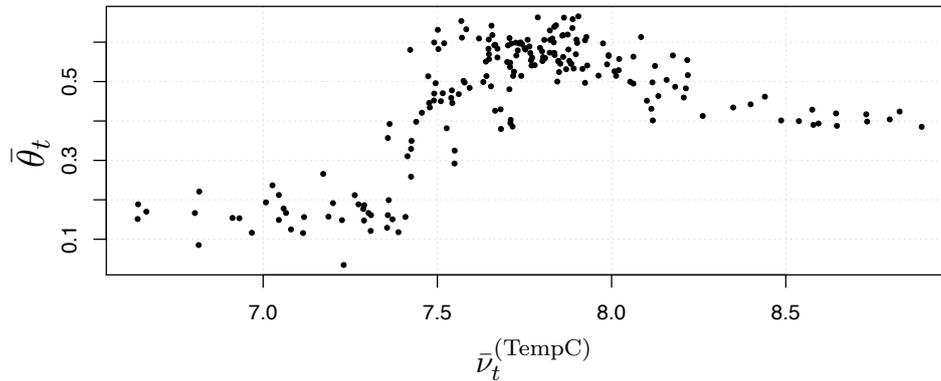
Figure 8.4: Plot for daily averages of temperature ($\bar{\nu}_t^{(\text{TempC})}$) against daily averages for polyp activity ($\bar{\theta}_t$). The estimated Pearson correlation coefficient between both is 0.54. A general connection between $\bar{\nu}_t^{(\text{TempC})}$ and $\bar{\theta}_t$ is assumed.

a similar way with exception of region $q = 10$. This region shows only a limited number of polyps and is considered as an outlier, here. A colonial coordination is therefore assumed. The effect of colonial coordination was also reported in a laboratory study [Price et al., 2017].

θ_t and ξ_t are correlated. This correlation is higher for the daily averages $\bar{\theta}_t$ and $\bar{\xi}_t$. Thus, the long-term change (about half a year) shows more significant functional relation. The coral color is shifting towards red and at about the same time the overall PA is increasing. The shifting back of the color starts about the same time when the PA decreases again. This effect was discussed with Pål Buhl-Mostensen who is an expert for *Lophelia pertusa*. It is expected that the food supply for the corals is changing throughout the year. During winter less food is expected to be available resulting in a lower PA. Furthermore, the type of food available is changing from phytoplankton to zooplankton which might by the reason for the color change of the coral [Buhl-Mortensen,P, personal communication, June, 2017].

The Pearson correlation coefficients for θ_t and the scalar sensor data f_t are highest for the conductivity sensor. Looking at Figure 8.1 this can be explained by seasonal effects, which are also noticeable for temperature and salinity and might not be a direct response of the polyps to changes of conductivity, temperature or salinity. Increased correlation coefficients for daily averages also reflect this effect. For the temperature, for instance, it seems that the coral has a preferred temperature window (see Figure 8.4). Polyp activity is highest for temperatures between $\approx 7.25 - 8.25$, but stating from this analysis that this is the optimal temperature window would need a more sophisticated analysis over longer time periods at different *Lophelia pertusa* coral reefs. For the chlorophyll sensor a medium correlation ($\text{corr}(\nu_t^{(\text{Chl})}, \theta_t) = -0.56$, $\text{corr}(\bar{\nu}_t^{(\text{Chl})}, \bar{\theta}_t) = -0.61$) with PA can be reported. Interesting here is that the correlation is negative, an increasing Chlorophyll seems to result in a decrease of θ_t. A higher Chlorophyll value is expected to trigger the PA as more food should be available. In the analyzed period an algae bloom event was detected resulting in very high Chlorophyll concentrations (see button

(a) wavelet response for θ_t

(b) wavelet response for $\nu_t^{(\mathrm{Vn_1})}$
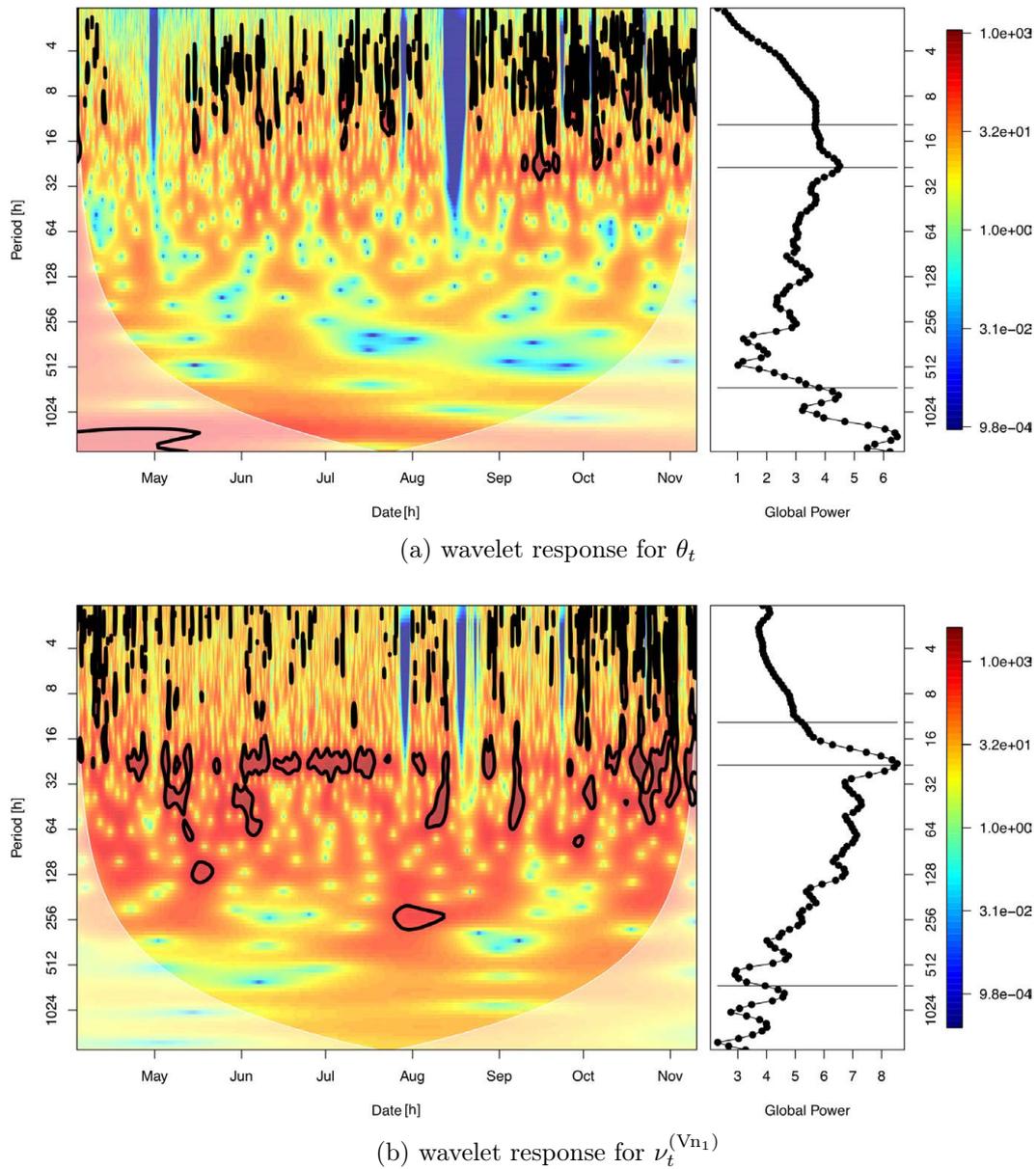
Figure 8.5: The response of the wavelet transformation for polyp activity θ_t (a) and current velocity in north direction $\nu_t^{(\mathrm{Vn_1})}$ (b). The corresponding color map on the left encodes the wavelet responses. Significant responses (significance level > 0.95) are highlighted by contours. The global power, shown as a plot on the right, visualizes the accumulated response for the entire period. Estimated tidal periodicity (12.4 h), 24 h periodicity and lunar periodicity ($\approx$ 707 h) are marked by black lines. The visualization (a) of wavelet responses for θ_t shows some time slots of low periodicity, which occur more frequently from September until mid November. In the plot of the global power a local maximum can be identified at a periodicity of 24 h for θ_t. In the visualization (b) of wavelet responses for $\nu_t^{(\mathrm{Vn_1})}$, several periods with a 24 h periodicity can be identified. The 24 h periodicity is also the strongest response in the global power.
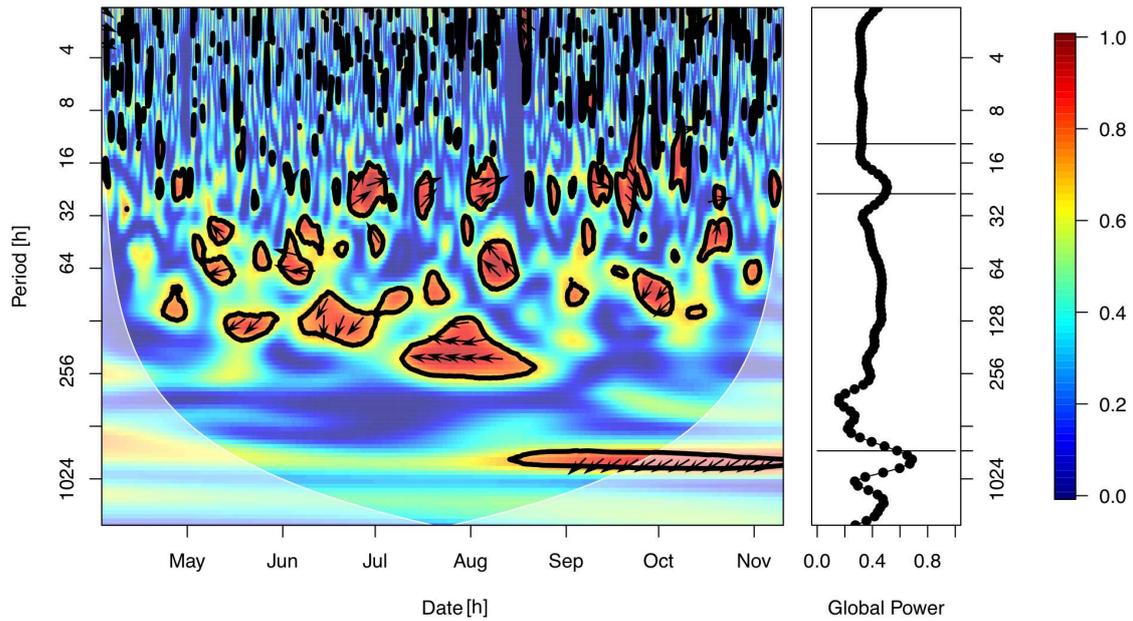
Figure 8.6: The response of the wavelet coherence analysis of polyp activity (θ'_t) and current velocity in north direction $(\nu_t^{'(\mathrm{Vn1})})$. The color map encodes the coherence significance of both time series [0,1]. A value of 1 would indicate a perfect coherence, while a value of 0 would indicate no coherence. Outlined areas mark a significance level > 0.95. The power plot on the right visualizes the average response for the entire period. Estimated tidal periodicity (12.4 h), 24 h periodicity and lunar periodicity ($\approx$ 707 h) are marked by black lines. Several periods with high coherence can be identified close to a 24 h periodicity, which is also noticeable as a local maximum in the global power plot (on the right). Highest global power is reached for a periodicity of 790 h $\approx$ 33 days.

graph in Figure 8.1). This event is located at the beginning of the analyzed time period featuring a low PA. Furthermore, the highest Chlorophyll concentrations are also influencing the images. Image color is shifted and also water turbidity is so high that coral polyps are not visible, resulting the in lowest PA estimates. Correlation coefficients for current velocities and PA are rather low, as current velocities do not show the seasonal trend like the other time series. A linear correlation analysis does not unveil the functional relationships here.

Wavelet responses were computed for all time series. The responses for θ_t (see Figure 8.5a) show a clear 24 h pattern, which is probably one of the most interesting findings here. Responses for shorter periodicities do most often occur from September until mid November, which is the end of the analyzed period. Those peaks in the wavelet transformation responses feature only very short durations. The polyp behavior is changing between short and 24 h periodicities. The 24 h periodicity could be the normal fall-back rhythm of the polyps and shorter periodicities might be caused by other external factors like the current velocity (see below). The wavelet responses for all current velocities at the lowest bin (2 m $- 7$ m above the sea-floor) show the strongest response (global power) for the 24 h periodicity, which is most clear for the northward direction (see Figure 8.5b). Current velocities seem to be independent from seasonal influences, or at least the effect of seasonal influences is rather low.

Wavelet coherence is only presented for $(\theta_t', \nu_t'^{(\mathrm{Vn1})})$, as this analysis showed the most significant results. Highest significance for coherence can be reported for the 24 h and 33 days periodicities. The 24 h periodicity can also be noticed in the responses of wavelet transformations for the individual time series θ_t and $\nu_t^{(\mathrm{Vn_1})}$). A coherence of PA and current velocity in north direction can therefore be reported. Polyps react to changing currents indicated by the arrows pointing in right direction for periods of 24 h in Figure 8.6. The 33 days periodicity, which is close to the moon phase periodicity (≈ 29.5 days) shows a mush lower response for $\nu_t^{(\mathrm{Vn_1})}$ than for θ_t. Nevertheless a coherence is noticeable. Extending the analyzed time period to multiple years might unveil this coherence with an increased significance. A correlation of the PA with the moon phase would be of interest as coral reproduction seems to be related to the moon phase as cold water corals tend to sperm only during full moon phase [Jarnegren,J , personal communication, June, 2017].

8.1.5   Conclusion

The presented workflow allowed for the first time to investigate function relationships between the visual feature of *Lophelia pertusa* extracted (semi-) automatically from images and other sensor data recorded by a stationary underwater observatory. This includes the inter-correlation analysis of PA, the correlation analysis with the coral color and the analysis of the functional relations with the other sensors of the LoVe observatory. The inter-correlation analysis of the PA demonstrates a possible colonial coordination of the *Lophelia pertusa in-situ*, which was previously only observed in a laboratory study. Furthermore, color change and PA seem to be connected. A possible biologically explanation for this might be the type and

amount of available food. A higher food supply might cause a higher feeding activity. In the log-term time scale (about half a year) a correlation between temperature and PA could be unveiled which of course might be just a derivative connection. Temperature changes seem to be related to seasonal fluctuations and this also accounts for the food supply. Relating the PA for over half a year with a temporal resolution of 1 h per image and sensor data unveiled a 24 h rhythm which showed a coherence with the current speeds. A full decomposition of all functional relations could not be applied. The analyzed time period is too short to unveil all the overlapping functional relations. Expanding the analyzed time period to several years with several FUO monitoring different sides of *Lophelia pertusa* coral reefs, would give the opportunity to apply additional studies to unveil more of those complex relations.

8.2 Wet-lab: Mesophyllum engelhartii

In order to generate models for responses of the environment to anthropogenic activities, laboratory studies are needed to identify key variables of the environment. The *wet-lab* study introduced in Chapter 7.1 is such a study. Functional relations between the treatment variables and the response variables need to be unveiled. The response variables include the visual features of the calcareous algae (CA) samples extracted from images in Chapter 7.1. Furthermore, the functional relations between the photosynthetic efficiency P measured during the experiment and the computed calcareous algae color $\bar{\mathbf{c}}_n^{(\mathrm{CA})}$ is investigated.

In this section the temporal analysis of the parameters is reduced, as the temporal resolution and duration is different. The investigated time period is only 10 weeks long and the temporal resolution is one image per week. It can be expected that time lag responses in the range of hours cannot be investigated as they will not be reflected in the data. The experiment was optimized in order to be able to generate a model, so the treatment variables could be optimized to cover a reasonable variation. Therefore, additionally to linear correlation analysis also a PLS regression is applied here in order to generate a model for the responses.

8.2.1 Material

The *wet-lab* experiment was conducted to investigate stress levels of CA induced by drill cuttings and explained briefly already in Section 7.1.1. CA samples are grouped in chambers where all samples in the same chamber are exposed to the same treatment parameters. The treatment parameters are light level, flow rates and added amount of sediment. CA responses were investigated weekly over seven weeks, by measuring sediment coverage (SC), photosynthetic efficiency (P) and recording digital images of the samples.

In Chapter 7.1 visual features, relevant for health status of investigated species *Mesophyllum engelhartii*, were extracted from images recorded during a wet-lab experiment. Extracted features were computed over all pixels segmented as live/stressed CA, i.e. the pixel positions marked in the corresponding B_n. Each image displays two CA covered rodoliths, i.e. two samples. Features are therefore computed for two samples. Visual features are relative size $\hat{A}_n$ and average color $\bar{\mathbf{c}}_n^{(\mathrm{CA})}$ (see Chapter 7.1 for details).

These visual features together with photosynthetic efficiency (P) and sediment coverage (SC), estimated during the experiment, are the response variables. The latter of course is not measuring a CA response and could also be handled as predictor variable, but due to the experimental setup it is also handled as a response variable. The purpose was to be able to predict the sediment coverage from the added amount of sediment and flow speed. All calcareous algae samples from the same chamber are exposed to the same treatment parameters. On each image two samples were captured. To avoid the need of a co-location of the samples as well as taking into account that all rhodolites are different, averages of the response variables within each chamber are used. Measurements for three different combinations of predictor variables for P and SC are illustrated in Figure 8.7.
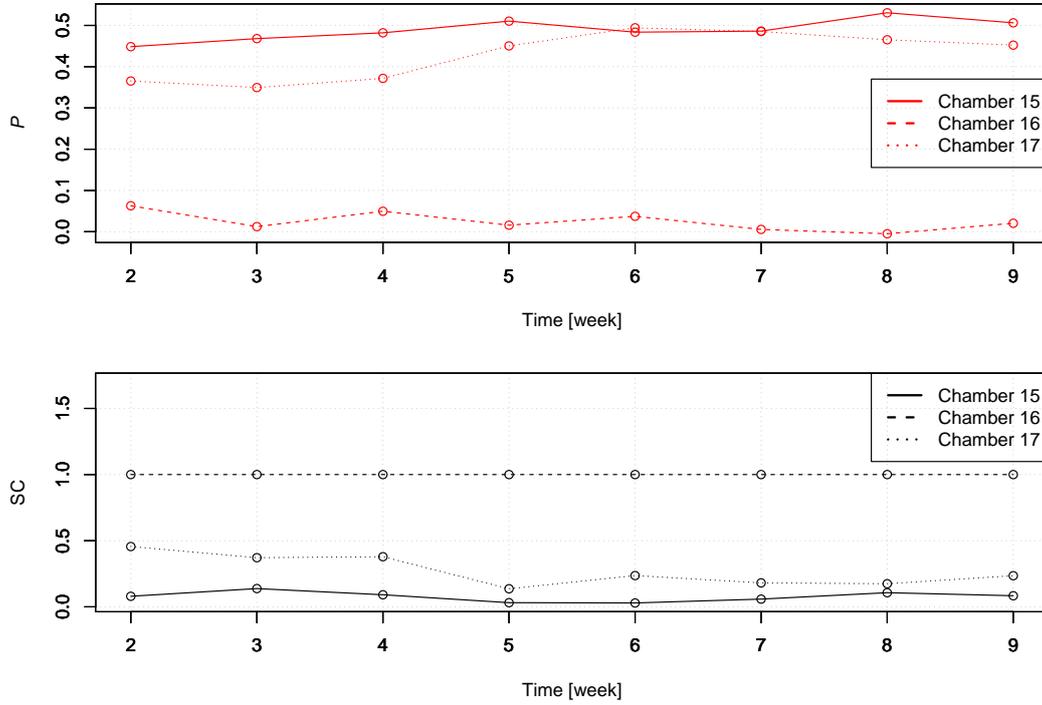
Figure 8.7: Photosynthetic efficiency P and sediment coverage SC for different combinations are plotted for chambers 15-17. All chamber feature different $\mathcal{S}$ concentration at highest $\mathcal{F}$ (for details see Table 7.1).

8.2.2 Methods

Several questions need to be answered here. First, is there a functional relation between photosynthetic efficiency and the extracted CA color. Second, can the treatment parameters be used to predict the CA response variables.

Photosynthetic efficiency vs. color In order to investigate the functional relation between $\bar{\mathbf{c}}_n^{(CA)}$ and P, Pearson correlation coefficients between both are computed. The extracted CA color $\bar{\mathbf{c}}_n^{(CA)}$ is therefore transformed into HSV and L*a*b* color spaces (see Section 2.2 for details). Correlation coefficients are then computed pairwise between $(\bar{c}_{n,L*}^{(CA)}, \bar{c}_{n,a*}^{(CA)}, \bar{c}_{n,b*}^{(CA)}, \bar{c}_{n,\text{red}}^{(CA)}, \bar{c}_{n,\text{green}}^{(CA)}, ...)$ and P. Furthermore, multivariate regression is performed with partial least squares (PLS) [Wold et al., 2012; Forina et al., 2007] using the R package "plsdepot" version 0.1.17 [Sanchez, 2012]. Using the R package, the data are mean centered and scaled to unit variance. Derived models are validated via five-fold cross validation. Averages of
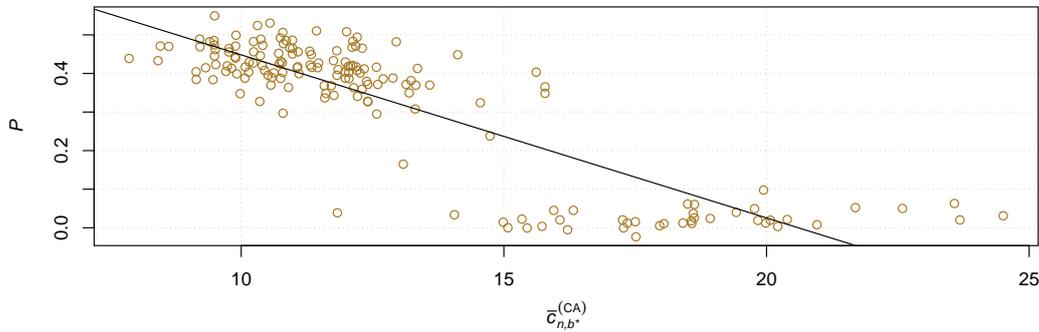
Figure 8.8: Plot for CA color (b*-channel) $\bar{c}_{n,b*}^{(CA)}$ against photosynthetic efficiency P. A Pearson regression coefficient of -0.86 between both variables was computed. The regression line is plotted in black.

squared Pearson correlation coefficients, i.e. coefficients of determination, for fitted (training) data and predicted (test) data are computed as goodness of fit and goodness of prediction, respectively. PLS-regression is applied to correlate multiple predictor values, in this case the color vector $\bar{\mathbf{c}}_n^{(CA)}$ transformed into several color spaces, to responses values, in this case P.

Treatment vs. CA responses Finally, in order to investigate the possible correlations of treatment and all response variables of the CA several statistical analysis are performed. First, multiple analysis of variance (ANOVA) tests and one multivariate analysis of variance (MANOVA) tests are conducted, followed by a pairwise correlation analysis. ANOVA and MANOVA tests are conducted to investigate the most general functional relation between predictor and response variables. Furthermore, as the predictor variables are discrete variables (only three different values for each variable) and can directly be used to group the responses, ANOVA and MANOVA tests are applied to investigate those group differences.

Second, multivariate regression is again performed with PLS-regression as described earlier to unveil multivariate relations. First, pairwise correlation analysis is conducted by computing the Pearson correlation coefficients. Second, multivariate regression is performed again using PLS-regression, as described earlier, to unveil multivariate relations.

8.2.3 Results

Photosynthetic efficiency vs. color First, the functional relation of CA color $\bar{\mathbf{c}}_n^{(CA)}$ and photosynthetic efficiency P was investigated. Pearson correlation coefficients scored highest for $b*$-channel of the $L*a*b*$ color space ($\mathrm{corr}(\bar{c}_{n,b*}^{(CA)}, P) = -0.86$, $\mathrm{corr}^2(\bar{c}_{n,b*}^{(CA)}, P) = 0.74$) and hue of the HSV color space ($\mathrm{corr}(\bar{c}_{n,\mathsf{hue}}^{(CA)}, P) = -0.80$, $\mathrm{corr}^2(\bar{c}_{n,\mathsf{hue}}^{(CA)}, P) = 0.65$). This high

Table 8.4: Results of pairwise ANOVA for the experiment data. Pairwise ANOVA p-values are displayed. Significant p-values are highlighted.

	$\hat{A}$	$\bar{c}_{n,L*}^{(CA)}$	$\bar{c}_{n,a*}^{(CA)}$	$\bar{c}_{n,b*}^{(CA)}$	P	SC
$\mathcal{F}$	0.5139	0.5715	0.0526	0.4010	0.3292	0.2920
$\mathcal{L}$	**0.0001**	0.2863	0.6570	0.0663	0.0223	0.2684
$\mathcal{S}$	0.7785	**< 0.0001**	**0.0003**	**< 0.0001**	**< 0.0001**	**< 0.0001**
$\mathcal{T}$	**< 0.0001**	0.1215	0.0008	**< 0.0001**	0.0556	0.9626

correlation is also visualized in Figure 8.8 for the b*-channel. The computed PLS-regression modeling P by $\bar{\mathbf{c}}_n^{(CA)}$ also scored highest with $\bar{\mathbf{c}}_n^{(CA)}$ transformed to the L*a*b* color space. This model scored an average goodness of fit of 0.78 and an average goodness of prediction of 0.75.

Treatment vs. CA responses For the sake of compactness, color responses of the CA samples are reduced to L*a*b* color responses, as they show the highest functional relations with the photosynthetic efficiency P (Pearson correlation and PLS-regression analysis). In order to investigate the possible correlations of treatment and all response variables of the CA samples, several ANOVA's (Table 8.4) and a MANOVA (Table 8.5) have been conducted. For the ANOVAs significant p-values have been estimated for all pairs of the treatment parameter $\mathcal{S}$ and all responses except for $\hat{A}$, the size of the CA. The flow $\mathcal{F}$ shows no significance for all response variables and the light $\mathcal{L}$ shows only a limited significance for $\hat{A}$. For the time $\mathcal{T}$ only $\hat{A}$ and $\bar{c}_{n,a*}^{(CA)}$ have significant p-values.

For the MANOVA significant p-values have been computed for all single treatment variables and for combinations of $\mathcal{F}$ and $\mathcal{L}$, $\mathcal{L}$ and $\mathcal{S}$, $\mathcal{S}$ and $\mathcal{T}$ and the triplet of $\mathcal{F}$, $\mathcal{L}$ and $\mathcal{S}$.

Next, pairwise regression has been computed (Table 8.6). It illustrates the importance of the variables although the coefficients are only based on pairwise regression. Highest coefficients of determination (corr^2) was estimated for pairs of $\mathcal{S}$ and P, $\mathcal{S}$ and SC. Sediment coverage is no direct CA response variable. It might therefore also be considered as a predictor variable. Thus, a pairwise regression was also conducted with SC and all other response variables. Here, highest coefficients of determination were estimated for $\bar{c}_{n,b*}^{(CA)}$ and P. It is also noticeable that $\mathcal{T}$ shows only reasonable correlation with $\hat{A}$. The pairwise regression coefficients have to be interpreted with care, as the treatment variables are discrete with only three different values each. Furthermore, they do not reflect multivariate relations.

Finally, PLS-regression was applied. Two PLS-regression models have been computed. The first (PLS$_1$), describes the relationship between all treatment variables, i.e. the predictors, $\mathcal{L}$, $\mathcal{F}$, $\mathcal{S}$, $\mathcal{S}^2$ and $\mathcal{T}$ and the response variables $\hat{A}$, $\bar{c}_{n,L*}^{(CA)}$, $\bar{c}_{n,a*}^{(CA)}$, $\bar{c}_{n,b*}^{(CA)}$, P, SC. It was obtained after expansion with one square term ($\mathcal{S}^2$). The second (PLS$_2$), is using only $\mathcal{T}$ and SC as predictors. It should be noticed that SC was used as a response variable for PLS$_1$.

Table 8.5: The results of MANOVA as p-values for different combinations of treatment variables. Values are computed using Wilks' Lambda. Significant p-values are highlighted.

Treatment variable(s)	p-value
$\mathcal{F}$	**< 0.0001**
$\mathcal{L}$	**< 0.0001**
$\mathcal{S}$	**< 0.0001**
$\mathcal{T}$	**< 0.0001**
$\mathcal{F} * \mathcal{L}$	**< 0.0001**
$\mathcal{F} * \mathcal{S}$	0.0013
$\mathcal{L} * \mathcal{S}$	**0.0002**
$\mathcal{F} * \mathcal{T}$	0.6991
$\mathcal{L} * \mathcal{T}$	0.0053
$\mathcal{S} * \mathcal{T}$	**< 0.0001**
$\mathcal{F} * \mathcal{L} * \mathcal{S}$	**< 0.0001**
$\mathcal{F} * \mathcal{L} * \mathcal{T}$	0.7413
$\mathcal{F} * \mathcal{S} * \mathcal{T}$	0.1933
$\mathcal{L} * \mathcal{S} * \mathcal{T}$	0.0170
$\mathcal{F} * \mathcal{L} * \mathcal{S} * \mathcal{T}$	0.2122

$*$ denotes the interaction between the treatment variables

Table 8.6: Pairwise variable correlation analysis of treatment parameters and CA responses. Coefficients of determination, as the squared Pearson correlation coefficient (corr^2), are displayed. Values > 0.5 are highlighted.

	$\hat{A}$	$\bar{c}_{n,L*}^{(CA)}$	$\bar{c}_{n,a*}^{(CA)}$	$\bar{c}_{n,b*}^{(CA)}$	P	SC
$\mathcal{F}$	0.00	0.00	0.02	0.00	0.01	0.00
$\mathcal{L}$	0.08	0.01	0.00	0.02	0.03	0.00
$\mathcal{S}$	0.00	0.35	0.07	0.47	**0.64**	**0.61**
$\mathcal{T}$	0.42	0.01	0.07	0.12	0.02	0.00
SC	0.00	0.41	0.07	**0.69**	**0.85**	—

Table 8.7: Goodness of fit and prediction of the PLS-regressions PLS_1 and PLS_2 (see text for details). Goodness of fit and prediction for the response variables as function of the four predictor variables $\mathcal{L}$, $\mathcal{F}$, $\mathcal{S}$ and $\mathcal{T}$ andSC, respectively.

	$\hat{A}$	$L*$	$a*$	$b*$	P	SC
PLS_1 fit.	**0.53**	0.45	0.18	**0.86**	**0.95**	**0.93**
PLS_2 fit.	0.42	0.43	0.13	**0.80**	**0.87**	-
PLS_1 pred.	**0.51**	0.49	0.16	**0.85**	**0.94**	**0.93**
PLS_2 pred.	0.43	0.46	0.15	**0.81**	**0.86**	-
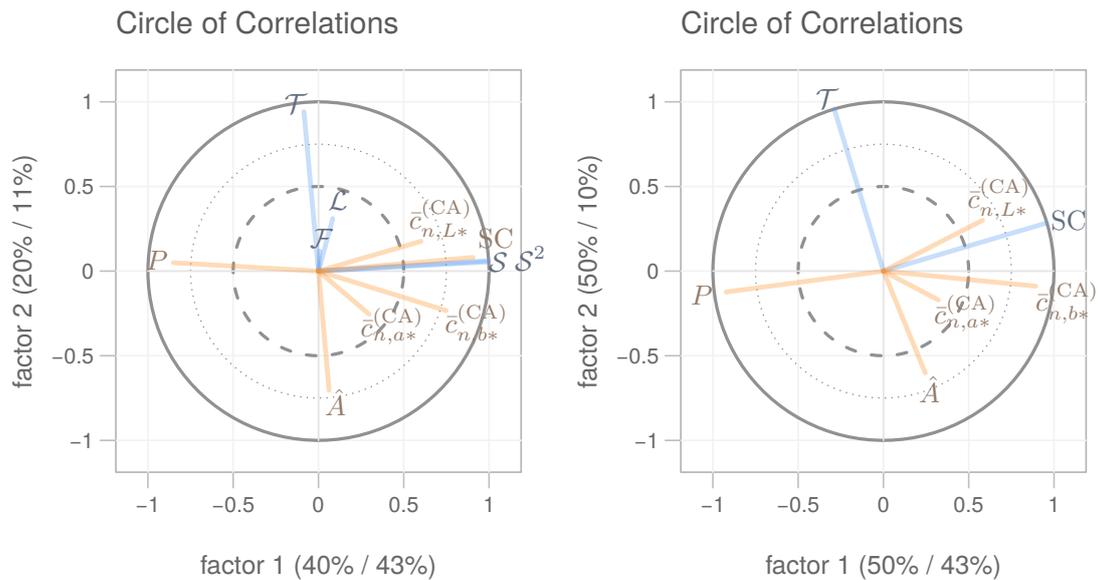
Figure 8.9: The correlation loading plots obtained from the PLS_1 (left) and PLS_2 (right). The outer circle represents 100% explained variance, the dotted circle represents 75% explained variance and the inner dashed circle represents 50% explained variance for the different variables projected on factor 1 and 2. Percentage values (prediction / fit) on the y and x-axes refer the explained variances of factor 1 and 2, respectively. PLS_1 was computed with light ($\mathcal{L}$), flow rate ($\mathcal{F}$), amount of sediment ($\mathcal{S}$, $\mathcal{S}^2$), and time ($\mathcal{T}$) as predictors highlighted in blue, the CA color responses ($\bar{c}_{n,L*}^{(CA)}$, $\bar{c}_{n,a*}^{(CA)}$), relative size ($\hat{A}$), photosynthetic efficiency (P) and sediment coverage (SC) as responses highlighted in red. PLS_2 was computed with only time $\mathcal{T}$ and sediment coverage SC as predictors and CA color ($\bar{c}_{n,L*}^{(CA)}$, $\bar{c}_{n,a*}^{(CA)}$), CA size ($\hat{A}$) and photosynthetic efficiency (P) as response variables.

Looking at the correlation plots of PLS_1 (Figure 8.9 on the left) and PLS_2 (Figure 8.9 on the right) some similarities can be noticed. For both time $\mathcal{T}$ is negatively correlated with $\hat{A}$ and $\mathcal{T}$ shows only very little correlation with the other responses. Furthermore, in both P is negatively correlated with the amount of sediment ($\mathcal{S}$) and sediment coverage (SC), respectively. CA color responses are correlated with $\mathcal{S}$ and SC, respectively. Time ($\mathcal{T}$) has only a limited influence on the colors. Light ($\mathcal{L}$) and flow ($\mathcal{F}$) were only used in PLS_1 and have only a very limited functional relation with all response variable where $\mathcal{F}$ in particular has the lowest influence.

Goodness of fit and prediction for both PLS-regressions, where prediction was obtained after cross validation, are summarized in Table 8.7. PLS_1 shows high goodness of fit (0.86) and prediction (0.85) for the b*-channel of the CA color response ($\bar{c}_{n,b*}^{(CA)}$), photosynthetic efficiency (P) and sediment coverage (SC). Size shows still a reasonable goodness of fit and prediction (> 0.5), while the a*-channel shows the lowest values here. Goodness of fit and prediction for PLS_2 also shows high goodness of fit and prediction for $\bar{c}_{n,b*}^{(CA)}$ and P, although estimated values are lower (Table 8.7). Goodness of fit and prediction for $\hat{A}$ is below 0.5.

8.2.4 Discussion

The investigation of the functional relation between CA color and photosynthetic efficiency unveiled a linear correlation between $\bar{c}_{n,b*}^{(CA)}$ and P (corr$^2 = 0.64$). Besides the biological interpretation of this effect, the reason for this correlation might be rather simple. The camera might have measured P indirectly. The b*-channels variation records a close or related wavelength also used by the fluorometer. In Figure 8.8 this linear correlation is illustrated and shown that a non-linear model should be more suitable. Additional experiments are needed in order to compute more complex and accurate models. Nevertheless, the PLS-regression showed that P is correlated (goodness of fit of 0.77) to the L*a*b*-values, implying that in principle it is possible to predict P from the color. The color extraction presented in Chapter 7.1 might be used alternatively to the direct measurement of P. This is of high importance for monitoring CA in the field, as photosynthetic efficiency measurement is challenging, if not impossible to conduct in-situ at ≈ 100 m depth.

Several analyses were conducted to investigate the impact of the treatment variables on the CA responses. The result of ANOVA analysis already showed that the amount of sediment ($\mathcal{S}$) might have the biggest impact on the CA stress level, noticeable as a change of photosynthetic efficiency (P), CA color $\bar{c}_n^{(CA)}$, but not the CA size $\hat{A}$. The MANOVA analysis confirmed this finding, but also showed that looking at pairs of treatment variables most variance can be explained by $\mathcal{S}$ and time ($\mathcal{T}$). Looking at MANOVAs for triplet or all response variables only the triple of $\mathcal{F}$, $\mathcal{L}$ and $\mathcal{S}$ has significance. Most of the variance in the responses can therefore be explained by only a few treatment parameters, in particular $\mathcal{S}$ and $\mathcal{T}$.

The first PLS-regression (PLS_1) demonstrates that in principle a modeling of the CA responses is possible here. It also shows that only the CA size is correlated with time, although the
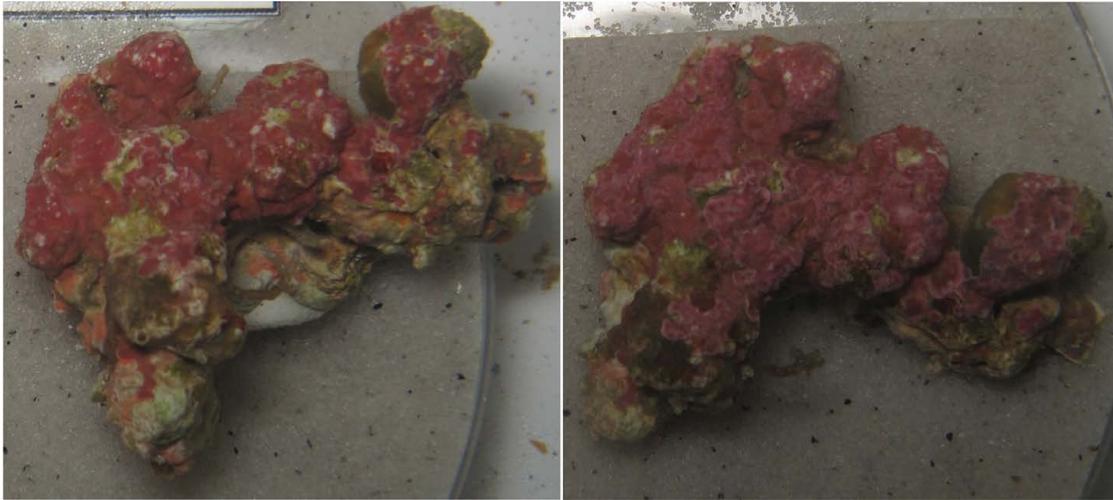
Figure 8.10: Positioning challenge example. Both images are displaying the same CA sample, but were recorded at different time points during the experiment. The CA sample was rotated around an axis not perpendicular to the image plane.

correlation is relatively weak. According to Table 8.7 goodness of fit and prediction is ≈ 0.5 for $\hat{A}$ as a function of all predictor variables, including time. Performing linear regression with $\hat{A}$ as a function of time alone gave a correlation coefficient of ≈ 0.4. The low correlation and the low goodness of fit might be caused by rotations of rhodolites. As illustrated in Figure 8.10, showing images recorded of the same rhodolith at two different time points. The rhodolite is rotated around an axis not perpendicular to the image plane 8.10. There, the functional relation between $\mathcal{T}$ and $\hat{A}$ might be higher than estimated. The PLS-regression analysis confirmed the findings of the ANOVAs, MANOVAs and linear regressions.

The second PLS-regression (PLS_2) was conducted to simulate a scenario comparable to field studies. In principle all those predictor variables can be estimated *in-situ* using a camera equipped FUO. It can be expected that the methods presented in Chapter 7.1 would need to be adapted for an *in-situ* application. In principle, a color monitoring is possible with images from an FUO, as illustrated in Section 6.1. As mentioned earlier, photosynthetic efficiency is a good stress indicator for CA. It could be shown that imaging can be used to predict the photosynthetic efficiency by extracting color information of imaged CAs. Furthermore, knowing the sediment coverage, which also might be extracted from time-lapse images, and the time of coverage also the photosynthetic efficiency can be predicted. Together this would allow to monitor CA samples *in-situ* of covered and uncovered samples. In Figure 8.11 such an FUO image is presented. Individual CA covered rhodolites are hard or even impossible to identify. Furthermore, as rhodolites roll around on the seafloor, a tracking of individuals, which is needed for instance to estimate time of coverage, is most probably impossible in these images. *In-situ* monitoring of CA still remains a highly challenging task.
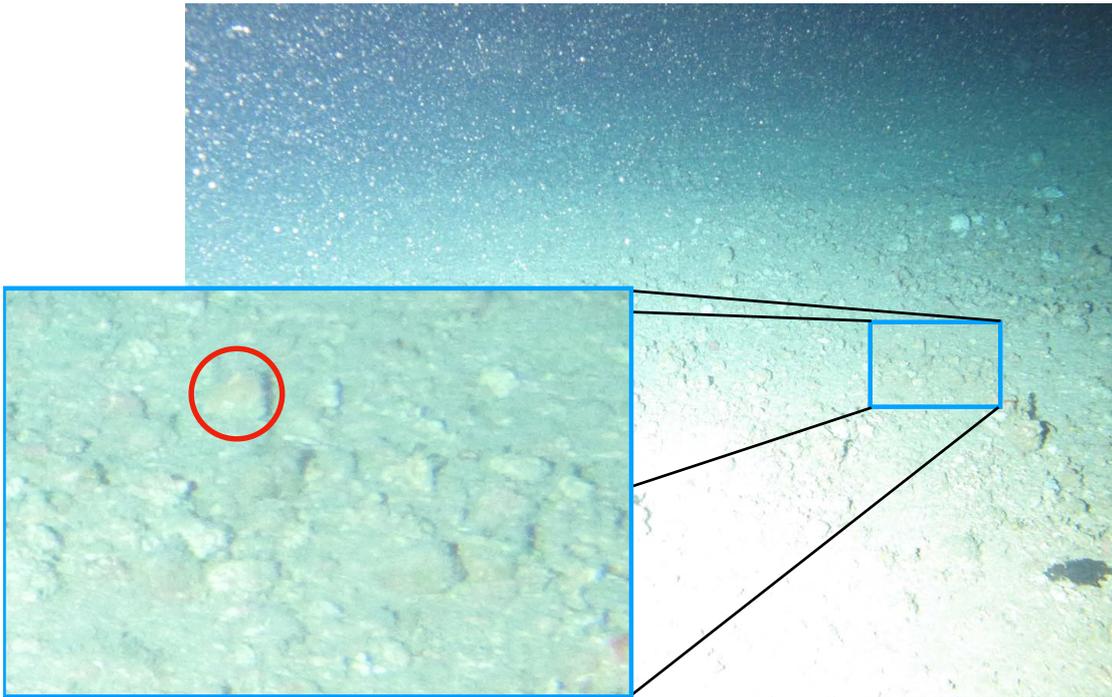
Figure 8.11: In-situ image for CA health monitoring. This image was taken by an FUO at the Perogrino oil field off the coast of Brazil. It illustrates that *in-situ* monitoring can still be a highly challenging task, as individual CA covered rhodolites are hard to identify in the presented image. A possible CA covered Rhodolite is highlighted in red.

8.2.5 Conclusion

The presented workflow allowed to investigate functional relationships between visual features of *Mesophyllum engelhartii* covered rhodolites, extracted (semi-) automatically from digital images recorded for a laboratory study. It could be shown that in principle photosynthetic efficiency measurements can be replaced by color measurements of the CA samples using image acquisition. In contrast to the *Lophelia pertusa* monitoring not the a*-color channel of the L*a*b* color space, but the b*-color channel seems to be able to represent change of the health status of the investigated species.

9 Discussion

9.1 Computer vision systems

In the previous Sections 6 and 7 CV has been applied to extract biologically relevant information from images recorded underwater. Four different CV systems have been designed for the different tasks of species investigation. All CV systems had to overcome the image acquisition challenges (OP challenge, DOP challenge) and the annotation challenges (AC challenge, AQ challenge, AU challenge). An overview of the individual resolving is given in Table 9.1.

All systems have the fixed camera system in common. The object-camera distances are rather stable throughout the several image time-series at least for the sessile species investigations. This simplifies resolving the effect of the OP challenge for all systems including the mobile species investigation (for details see below).

For the color monitoring of a *Lophelia pertusa* coral reef, the most difficult challenge to be resolved was the DOP challenge. The color of the imaged cold water coral had to be monitored over time. The dynamics of the optical properties of the water can course a color change of the coral in the images. In this case rapid changes are not the biggest concern, but slow seasonal changes for instance caused by the presence or absence of organism or change of composition of the present plankton community. A color reference like the reference plate in Chapter 7.1 would allow to compensate those effects straightforward, but is absent in the images. The presented approach resolved this by using the non-coral area as a color reference in the images. The OP challenge was resolved by the use of a fixed camera and a fixed white balancing. Distances between the sessile OOI, i.e. the coral reef, are stable throughout the analyzed time period as are the effects of the OP challenge. The AC challenge was resolved by the use of unsupervised ML and an example based annotation approach. The followed manual re-inspection and optimization resolved also the AQ challenge. The use of a fixed mask resolved the AU challenge. The low variation of the appearance of the coral simplified the segmentation task in general. The use of color-independent features finally allowed to achieve reasonable segmentation accuracy despite the change of color of the coral. A more accurate segmentation would only be for measuring the size of life coral over time.

For the polyp activity estimation the annotation challenges were the hardest part to be resolved. Thus, for this task a comprehensive annotation study was needed to be performed. The gradient of polyp activity used for the manual annotation was reduced to only two states: active and inactive, to resolve the AC challenge and the AQ challenge. The AC challenge was also resolved by data augmentation. The AU challenge affected mostly the evaluation and thus, a customized recall and FPR-like measurement were defined. Furthermore, a rank-correlation between manual and automated activity estimation was computed. The OP challenge was resolved by focusing on a well illuminated center section of the images for both the polyp activity estimation and the manual annotation. The DOP challenge was resolved by including images with higher turbidity in the annotation study and thus also in

the training of the CNN. Nevertheless, it could not completely be resolved. For very low visibilities, for instance during an algae bloom event, the polyp activity estimation failed. The DOP challenge can not be solved in this case after the image acquisition.

For the shrimp identification and localization the most challenging part is the semi-transparency of the species instances. The effect when using fixed cameras is in a way similar to the OP challenge. Shrimp colors are dependent on the location within the image. This and the OP challenge where both resolved by the use of temporal color contrast features. Nevertheless, the detection rate is still dependent on the background. Shrimp are more likely to be identified on white live coral than on dark areas below the reef, but this also applies for the manual annotation. The use of temporal color contrast features also resolved the DOP challenge at least partly. Rapid changes of the matter in the water column will still affect the image colors but slow changes can be compensated as the background is modeled dynamically. The computation of the shrimp contrast map could be further optimized for the purpose of resolving the DOP challenge, for instance by computing the mean image weighted by a Gaussian model also taking images $I_{t+1,...}$ recorded after the actual processed image I_t into account. The AQ challenge is also quite problematic for the shrimp detection approach, as it is very hard to produce consistent shrimp annotations. During the manual annotation, it could be observed that the confidence for annotating very small shrimp changed, as it increased. The use of super-pixels allowed to compensate for this effect. Furthermore, it allowed to speed up the manual annotation while limiting the effect of the AU challenge. The AC challenge also was partly resolved by the use of super-pixels and the use of point shaped annotations resulting in several annotation per shrimp. It was also resolved by taking all non-annotated super-pixels as the background class generating a highly unbalanced two class problem. Thus a classifier had to be selected which can handle such highly imbalanced training data.

The last CV system was designed to quantify color and size of live calcareous algae (CA). Due to the controlled environment of a *wet-lab* imaging conditions could be highly optimized. Placing a color reference plate next to the imaged OOI allowed to achieve a color constancy of high quality throughout the whole time period of the experiment. The OP challenge was resolved by evaluating the imaged color reference plate for the preprocessing. Furthermore, the stable camera-object distance reduced the variation of possible in water image acquisition effects between different images. Last but not least also the variation of the object-camera distance within an image is small compared to the *in-situ* imaging setup. The effect of the OP challenge might therefore in this case be negligible. Effects of the DOP challenge are noticeable in some images due to sediment dust in the water column. The use of the reference plate again was here the solution to this challenge. The use of a mixture of artificial light and natural light might have also affected the image colors and could also be resolved by reading the color reference plate. Similar to the *in-situ* color monitoring of the coral the AC challenge and AQ challenge were resolved by using an example based annotation strategy and unsupervised ML for the segmentation of the CA samples. The AU challenge could again be resolved by computing recall and FPR rates for the evaluation of the segmentation.

Table 9.1: Overview of all CV applications and their corresponding challenge resolving.

Project	Coral color	Polyp activity	Shrimp detection	Calcareous algae
OP challenge	• fixed white balancing	• fixed white balancing • focus on center area	• temporal color contrast features (TCCF)	• color reference plate • illumination correction
DOP challenge	• non-coral area as color reference	• training images with varying turbidity	• temporal color contrast features (TCCF)	• color reference plate • illumination correction
AC challenge	• example based annotation approach • unsupervised ML	• data augmentation • reduction of classification classes • automated selection of a background class	• multiple point annotations per OOI • non-shrimp as background class • ML for highly unbalanced training data	• example based annotation strategy • annotating each imaged CA • unsupervised ML
AQ challenge	• manual re-inspection and optimization	• comprehensive three step labeling study • reduction of classification classes	• size limitations using super-pixel • manual re-inspection	• example based annotation strategy • annotating each imaged CA • unsupervised ML
AU challenge	• constant segmentation mask	• circle shaped annotations • three step evaluation	• multiple point annotations per OOI • pre-segmentation with super-pixels	• evaluation with recall and FPR

9.2   Data science

The data science approach, creating new knowledge from the heterogeneous data in form of images and environmental sensor data was presented in Chapter 8 exemplarily for the coral monitoring and the CA response study. Both approaches use linear regression to unveil new functional coherences between behavioral indicators for the species of investigation and sensor measurements from the environment. Due to the different temporal resolutions and duration, further analyzes differ between both. Temporal resolution ($\frac{1}{h}$ vs. $\frac{1}{week}$) and duration ($\approx$ half a year vs. 10 weeks) are higher for the *in-situ* monitoring of the coral allowing to apply the analysis in the frequency domain via wavelet analysis. In order to unveil functional relations of combinations of sensor data and behavior information PLS regression was applied for the CA response study. In principle this could also be applied to the data of the color monitoring studies, but the PLS does not consider delays of responses. In Section 8.1 for instance a delay between current change and change of polyp activity was identified.

10   Conclusion

The focus in this thesis has been to create computational solutions for the corresponding biologically motivated research questions. The development of individually tailored CV systems for the different tasks allowed to use as much meta information as possible to increase the performance of the corresponding CV systems. This information includes the expected size variation of a species, the type of species, i.e. sessile or mobile, and the type of images and camera systems. The stable imaging setup of the LoVe FUO in particular allowed to resolve the image acquisition challenges for the different CV systems and was essential for the shrimp monitoring.

In the context of this thesis five challenges for marine CV have been identified: the image acquisition challenges (OP challenge, DOP challenge) and the annotation challenges (AC challenge, AQ challenge, AU challenge). All challenges have successfully been resolved individually for the different CV tasks of marine environmental monitoring including the computational analysis of a *wet-lab* experiment.

A detailed description of the process of manually annotating underwater images was presented in Chapter 3. In order to resolve some of the annotation challenges, a quality assessment for manual underwater annotations was described. This includes the creation of a gold standard and evaluation for correctness and reproducibility of manual annotations. The quality assessment is of high importance for a manual annotation of underwater images in general and especially for developing CV systems as the annotations provide the ground truth data for the machine learning.

In order to resolve the image acquisition issues after image recording, several algorithms have been introduced for color and illumination normalization in Chapter 5. Furthermore, a score was defined to rank different preprocessing methods by their ability to resolve the image acquisition challenges. Manual annotations are needed for this approach and the different OOI classes are evaluated for intra- and inter-class-variation for this ranking. It was developed mainly for applying CV, but it might also help to identify preprocessing methods which can be applied to increase the quality of manual annotations.

CV has been applied successfully in this thesis to investigate species behavior from images recored *in-situ* and for *wet-lab* experiments. In Section 6.1, for the first time a color change of the *Lophelia pertusa* could be observed and quantified *in situ*. This was possible without the need of a color reference plate and by the use of a standard consumer camera despite the OP challenge and DOP challenge. Furthermore, applying CV also allowed for the first time to assess the *in-situ* polyp behavior for a duration of over half a year with a temporal resolution of one hour per image. A seasonal trend of both color and polyp activity was observed. In Section 6.2, shrimp were identified and localized in LoVe images despite their semi-transparent appearance. The developed CV system could be applied to unveil a spatial distribution change comparing two different time periods with a three week duration. In Chapter 7, a CV system was successfully tailored to investigate stress levels of calcareous algae samples in a *wet-lab* experiment. The size and color change of the imaged samples

over time could thus be quantified using a consumer camera and a color reference plate placed in the field of view for each recorded image.

Generating CV systems is often only the first step to generate new relevant biologically information. The extracted image information (species behavior and color) needs to be related to other environmental parameters as done in Chapter 8. Data science methods allowed to identify a diel pattern (24 h) for the polyp activity, which is most likely connected to current changes (Section 8.1). Furthermore, it could be unveiled that the coral color is changing with the level of polyp activity. This effect is most probably caused by the consumption of a type of food containing the carotenoid astaxantin. The amount and type of food is connected to the coral behavior in form of color response and feeding activity.
Data science methods were applied to evaluate a *wet-lab* experiment (Section 8.2). In this study, a correlation between photosynthetic efficiency and color of calcareous algae samples was identified. These findings might allow a health status (coral bleaching) monitoring *in-situ* with standard cameras. Furthermore, a model was created for the impact of sedimentation on calcareous algae covered rhodoliths.

10.1   Future perspectives

The recently wide and rapid development of CV algorithms and deep learning applications in particular will also improve the performance of CV systems for marine environmental monitoring in the future. On the contrary, underwater CV is a separate research field with its own challenges like the ones identified in this thesis.

A promising attempt is the use of pre-trained networks as for instance applied in [Raitoharju et al., 2016]. Also, the use of networks specifically designed for time-series analysis like LSTM (long short term memory) networks might also increase the performance in future underwater CV applications for FUO images.

In order to overcome one of the biggest challenges, the bottleneck of manual annotation process the use of citizen science might be a solution. In [Matabos et al., 2017] promising results have been published demonstrating that crowds can also create reliable annotations depending on the required level of detail and complexity of the annotation problem. Also, computer assisted annotation system like [Möller et al., 2017] might speed up this process and increase the quality of the manual annotations. Such a computer assisted annotation allows to transpose the time demanding whole image screening (WIS) task into a faster single patch confirmation (SPC) task.

The color normalization for the *in-situ* monitoring of the *Lophelia pertusa* color (Section 6.1.2) would benefit from the presence of a real color reference. Also integrating other sensor for e.g. turbidity or chlorophyll in the color monitoring would allow to increase the reliability of the system. The latter also accounts for the polyp activity monitoring (Section 6.1.3). For high turbidities low polyp activity was detected by the current system. A more detailed analysis of this correlation would improve the reliability of the system. Furthermore, other deep learning algorithms which are able to use information of previous images, like (convolutional)

recurrent neural networks, might increase the performance and robustness of the system and might allow to resolve the DOP challenge in particular. Comparing different level of polyp activity for individual polyps was necessary for the manual annotation of the different level of activity. Including image information from previous images in a CV system therefore has the potential to increase the performance.

The identification and localization of the shrimp (Section 6.2) also might benefit from such deep learning algorithms. The TCCF should be encapsulated in a deep learning approach to further increase the performance of the CV system. For the calcareous algae monitoring, the impact of different concentrations of sediment should be evaluated further, as sediment coverage of the rhodolite samples is either fully or nearly uncovered. Furthermore, a pre-written guidance about the CV specific requirements to conduct such experiments should to be created. A rotation of the samples for the different image recordings for instance has to be avoided as suggested in [Osterloff et al., 2014a].

In addition to the improvements of the CV systems the data science part should be continued as the presented analysis is just a scratch on the surface. Investigating even longer time periods to identify possible seasonal or yearly variations should be carried out.

The LoVe observatory will acquire new data. It has been off-line since November 2015, but is now re-deployed at a different location again to monitor a cold water reef of *Lophelia pertusa*. Additionally to this LoVe FUO another FUO has been deployed, equipped with a stereo camera to monitor instances of bubblegum coral (*Paragorgia arborea*). Both FUO will allow to investigate similar species and new species of interest for longer time periods to unveil new knowledge about their behavior and the complex underwater environment they are living in. So for future investigation the presented systems most probably will need to be adapted to other FUOs including the new LoVe observatories. Furthermore, new CV systems will need to be developed to monitor different types of species. Behavior of other species has to be encapsulated in future data science approaches.

Bibliography

Achanta, R., A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk (2010). *SLIC Superpixels*. Tech. rep. EPFL.

Achanta, R., A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süssstrunk (Nov. 2012). "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11, pp. 2274–2282. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2012.120.

Aguzzi, J., A. Mànuel, F. Condal, J. Guillén, M. Nogueras, J. del Rio, C. Costa, P. Menesatti, P. Puig, F. Sardà, D. Toma, and A. Palanques (2011). "The new seafloor observatory (OBSEA) for remote and long-term coastal ecosystem monitoring". In: *Sensors* 11.6, pp. 5850–5872. ISSN: 14248220. DOI: 10.3390/s110605850.

Åhlén, J., D. Sundgren, and E. Bengtsson (2007). "Application of underwater hyperspectral data for color correction purposes". In: *Pattern Recognition and Image Analysis* 17.1, pp. 170–173.

Amado-Filho, G. M., R. L. Moura, A. C. Bastos, L. T. Salgado, P. Y. Sumida, A. Z. Guth, R. B. Francini-Filho, G. H. Pereira-Filho, D. P. Abrantes, P. S. Brasileiro, R. G. Bahia, R. N. Leal, L. Kaufman, J. a. Kleypas, M. Farina, and F. L. Thompson (2012). "Rhodolith beds are major CaCO 3 BIO-factories in the tropical south West Atlantic". In: *PLoS ONE* 7.4, e35171. ISSN: 19326203. DOI: 10.1371/journal.pone.0035171.

Ancuti, C. O., C. Ancuti, C. D. Vleeschouwer, and P. Bekaert (Jan. 2018). "Color Balance and Fusion for Underwater Image Enhancement". In: *IEEE Transactions on Image Processing* 27.1, pp. 379–393. ISSN: 1057-7149. DOI: 10.1109/TIP.2017.2759252.

Arnold-Bos, A., J.-P. Malkasse, and G. Kervern (2005). "A preprocessing framework for automatic underwater images denoising". In: *European Conference on Propagation and Systems* 1, p. 8. URL: http://hal.archives-ouvertes.fr/hal-00494314.

Baker, M. (2016). "1,500 scientists lift the lid on reproducibility". In: *Nature* 533.7604, pp. 452–454. DOI: 10.1038/533452a.

Barange, M. (2010). *Marine ecosystems and global change*. eng. Oxford biology. Oxford Univ. Press. ISBN: 978-0-19-955802-5, 978-0-19-960089-2.

Barnes, C. R., M. M. R. Best, F. R. Johnson, and B. Pirenne (2015). "NEPTUNE Canada: Installation and initial operation of the world's first regional cabled ocean observatory". In: *Seafloor Observatories: A New Vision of the Earth from the Abyss*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 415–438. ISBN: 9783642113741. DOI: 10.1007/978-3-642-11374-1_16.

Barron, J. T. (2016). "Convolutional color constancy". In: *Proceedings of the IEEE International Conference on Computer Vision*. Vol. 11-18-December-2015, pp. 379–387. ISBN: 9781467383912. DOI: 10.1109/ICCV.2015.51. arXiv: 1507.00410.

Barros, W., E. R. Nascimento, W. V. Barbosa, and M. F. Campos (2018). "Single-shot underwater image restoration: A visual quality-aware method based on light propagation model". In: *Journal of Visual Communication and Image Representation* 55, pp. 363 –373. ISSN: 1047-3203. DOI: 10.1016/j.jvcir.2018.06.018.

Basso, D. (1998). "Deep rhodolith distribution in the Pontian Islands, Italy: A model for the paleoecology of a temperate sea". In: *Palaeogeography, Palaeoclimatology, Palaeoecology* 137.1-2, pp. 173–187. ISSN: 00310182. DOI: `10.1016/S0031-0182(97)00099-0`.

Bayer, B. E. (1976). "Color imaging array". In: *US Patent, 3,971,065*.

Bazeille, S., I. Quidu, L. Jaulin, and J.-P. Malkasse (Oct. 2006). "Automatic underwater image pre-processing". In: *CMM'06*. Brest, France, p. xx. URL: `https://hal.archives-ouvertes.fr/hal-00504893`.

Beijbom, O., P. J. Edmunds, D. I. Kline, B. G. Mitchell, and D. Kriegman (June 2012). "Automated annotation of coral reef survey images". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1170–1177. DOI: `10.1109/CVPR.2012.6247798`.

Beijbom, O., T. Treibitz, D. I. Kline, G. Eyal, A. Khen, B. Neal, Y. Loya, B. G. Mitchell, and D. Kriegman (2016). "Improving Automated Annotation of Benthic Survey Images Using Wide-band Fluorescence". In: *Scientific Reports* 6.August 2015, p. 23166. ISSN: 2045-2322. DOI: `10.1038/srep23166`.

Bellman, R. E. (1961). *Adaptive control processes: a guided tour*. Princeton university press.

Benfield, M., P. Grosjean, P. Culverhouse, X. Irigolen, M. Sieracki, A. Lopez-Urrutia, H. Dam, Q. Hu, C. Davis, A. Hanson, C. Pilskaln, E. Riseman, H. Schulz, P. Utgoff, and G. Gorsky (2007). "RAPID: Research on Automated Plankton Identification". In: *Oceanography* 20.2, pp. 172–187. ISSN: 10428275. DOI: `10.5670/oceanog.2007.63`.

Bianco, G, M Muzzupappa, F Bruno, R Garcia, and L Neumann (2015). "A NEW COLOR CORRECTION METHOD FOR UNDERWATER IMAGING". In: *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XL-5/W5, pp. 25–32. DOI: `10.5194/isprsarchives-XL-5-W5-25-2015`.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer. ISBN: 978-0387310732.

Box, G. E. P., W. G. Hunter, and J. S. Hunter (1978). "Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building". In: *Wiley Series in Probability and Mathematical Statistics* 1st, p. 653.

Breiman, L., J. Friedman, C. Stone, and R. Olshen (1984). *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis. ISBN: 9780412048418.

Breiman, L. (Oct. 2001). "Random Forests". In: *Machine Learning* 45.1, pp. 5–32. ISSN: 1573-0565. DOI: `10.1023/A:1010933404324`.

Brewer, C. A. (1999). "Color Use Guidelines for Data Representation". In: *Proceedings of the Section of Statistical Graphics*, pp. 55–60. ISSN: 1483287920. DOI: `10.1016/B978-0-08-042415-6.50013-2`.

– (2013). *ColorBrewer: Color Advice for Maps*. URL: `http://colorbrewer2.org` (visited on 12/16/2015).

Bryson, M., M. Johnson-Roberson, O. Pizarro, and S. Williams (June 2012). "Colour-Consistent Structure-from-Motion Models using Underwater Imagery". In: *Proceedings*

of Robotics: Science and Systems. Sydney, Australia. DOI: `10.15607/RSS.2012.VIII.005`.

Buchsbaum, G. (1980). "A spatial processor model for object colour perception". In: *Journal of the Franklin Institute* 310.1, pp. 1–26. DOI: `10.1016/0016-0032(80)90058-7`.

Buhl-Mortensen, P., E. Tenningen, and A. B. S. Tysseland (2015). "Effects of water flow and drilling waste exposure on polyp behaviour in Lophelia pertusa". In: *Marine Biology Research* 11.7, pp. 725–737.

Caliński, T. and J. Harabasz (1974). "A dendrite method for cluster analysis". In: *Communications in Statistics-theory and Methods* 3.1, pp. 1–27.

Canny, J. (1986). "A computational approach to edge detection." In: *IEEE transactions on pattern analysis and machine intelligence* 8.6, pp. 679–698. ISSN: 0162-8828. DOI: `10.1109/TPAMI.1986.4767851`.

Carlevaris-Bianco, N., A. Mohan, and R. M. Eustice (2010). "Initial results in underwater single image dehazing". In: *OCEANS 2010*. IEEE, pp. 1–8. DOI: `10.1109/OCEANS.2010.5664428`.

Chambah, M., D. Semani, A. Renouf, P. Courtellemont, and A. Rizzi (2003). "Underwater color constancy: enhancement of automatic live fish recognition". In: *Electronic Imaging 2004*. International Society for Optics and Photonics, pp. 157–168.

Chen, C. and L.-M. Liu (1993). "Joint Estimation of Model Parameters and Outlier Effects in Time Series". In: *Journal of the American Statistical Association* 88.421, pp. 284–297. ISSN: 01621459. DOI: `10.2307/2290724`.

Chiang, J. Y. and Y. C. Chen (Apr. 2012). "Underwater Image Enhancement by Wavelength Compensation and Dehazing". In: *IEEE Transactions on Image Processing* 21.4, pp. 1756–1769. ISSN: 1057-7149. DOI: `10.1109/TIP.2011.2179666`.

Coffin, D. (2016). *Decoding raw digital photos in Linux.* URL: `http://www.cybercom.net/~dcoffin/dcraw/` (visited on 03/13/2017).

Cortes, C. and V. Vapnik (Sept. 1995). "Support-vector networks". In: *Machine Learning* 20.3, pp. 273–297. ISSN: 1573-0565. DOI: `10.1007/BF00994018`.

Cristini, L., R. S. Lampitt, V. Cardin, E. Delory, P. Haugan, N. O'Neill, G. Petihakis, and H. A. Ruhl (Sept. 2016). "Cost and value of multidisciplinary fixed-point ocean observatories". In: *Marine Policy* 71, pp. 138–146. ISSN: 0308597X. DOI: `10.1016/j.marpol.2016.05.029`.

Davies, A. J., J. M. Roberts, and J. Hall-Spencer (2007). "Preserving deep-sea natural heritage: Emerging issues in offshore conservation and management". In: *Biological Conservation* 138.3-4, pp. 299–312. ISSN: 00063207. DOI: `10.1016/j.biocon.2007.05.011`.

Davies, D. L. and D. W. Bouldin (1979). "A cluster separation measure". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2, pp. 224–227.

Deng, L. and D. Yu (2014). "Deep Learning: Methods and Applications". In: *Foundations and Trends in Signal Processing* 7, pp. 197–387. ISSN: 1932-8346. DOI: `10.1561/2000000039`.

Dhar, V. (Dec. 2013). "Data Science and Prediction". In: *Commun. ACM* 56.12, pp. 64–73. ISSN: 0001-0782. DOI: `10.1145/2500499`.

Duntley, S. Q., A. R. Boileau, and R. W. Preisendorfer (June 1957). "Image Transmission by the Troposphere I∗". In: *J. Opt. Soc. Am.* 47.6, pp. 499–506. DOI: `10.1364/JOSA.47.000499`.

Durden, J. M., B. J. Bett, T. Schoening, K. J. Morris, T. W. Nattkemper, and H. A. Ruhl (2016a). "Comparison of image annotation data generated by multiple investigators for benthic ecology". In: *Marine Ecology Progress Series* 552, pp. 61–70. ISSN: 01718630. DOI: `10.3354/meps11775`.

Durden, J. M., T. Schoening, F. Althaus, A. Friedman, R. Garcia, A. G. Glover, J. Greinert, N. J. Stout, D. O. Jones, A. Jordt, J. W. Kaeli, K. Köser, L. A. Kuhnz, D. Lindsay, K. J. Morris, T. W. Nattkemper, J. Osterloff, H. A. Ruhl, H. Singh, M. Tran, and B. J. Bett (Nov. 2016b). "Perspectives in visual imaging for marine biology and ecology: from acquisition to understanding". In: *Oceanography and Marine Biology: An Annual Review, Vol. 54*. Ed. by R. Hughes, D. Hughes, I. Smith, and A. Dale. Boca Raton, FL: CRC Press, pp. 1–73. ISBN: 978-1-4987-4798-1. DOI: `10.1201/9781315368597-2`.

Ebner, M. (2007). *Color Constancy*. The Wiley-IS&T Series in Imaging Science and Technology. Wiley. ISBN: 9780470510483.

Erat, O., K. Panetta, and S. Agaian (Apr. 2017). "Contrast enhancement for underwater images in maritime border protection". In: *2017 IEEE International Symposium on Technologies for Homeland Security (HST)*, pp. 1–5. DOI: `10.1109/THS.2017.8050197`.

Evangelidis, G. and E. Psarakis (2008). "Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.10, pp. 1858–1865. ISSN: 0162-8828. DOI: `10.1109/TPAMI.2008.113`.

Felzenszwalb, P. F. and D. P. Huttenlocher (Sept. 2004). "Efficient Graph-Based Image Segmentation". In: *International Journal of Computer Vision* 59.2, pp. 167–181. ISSN: 1573-1405. DOI: `10.1023/B:VISI.0000022288.19776.77`.

Field, J. G. Â., G. Hempel, and C. P. Summerhayes, eds. (2002). *Oceans 2020*. eng. Washington [u.a.]: Island Press, XVIII, 365 S. : Ill., graph. Darst. ISBN: 1-55963-469-3, 1-55963-470-7.

Figueiredo, M. a. O., I. Eide, M. Reynier, A. B. Villas-Bôas, F. T. S. Tâmega, C. G. Ferreira, I. Nilssen, R. Coutinho, and S. l. Johnsen (2015). "The effect of sediment mimicking drill cuttings on deep water rhodoliths in a flow-through system: Experimental work and modeling". In: *Marine Pollution Bulletin* 95.1, pp. 81–88. ISSN: 0025-326X. DOI: `http://dx.doi.org/10.1016/j.marpolbul.2015.04.040`.

Fixed-Point Open Ocean Observatories (2015). *Observatories*. URL: `http://www.fixo3.eu/observatory/` (visited on 06/10/2015).

Forina, M, S Lanteri, and M Casale (2007). *Multivariate calibration*. Vol. 1158. John Wiley & Sons, pp. 61–93. ISBN: 0103532684. DOI: `10.1016/j.chroma.2007.03.082`.

Foster, M. S. (2001). "Rhodoliths: Between rocks and soft places". In: *Journal of Phycology* 37.5, pp. 659–667. ISSN: 00223646. DOI: `10.1046/j.1529-8817.2001.00195.x`.

Freiwald, A., J. B. Wilson, and R. Henrich (1999). "Grounding pleistocene icebergs shape recent deep-water coral reefs". In: *Sedimentary Geology* 125.1-2, pp. 1–8. ISSN: 00370738. DOI: `10.1016/S0037-0738(98)00142-0`.

Funk, C., S. Bryant, and P. Heckman Jr (1972). *Handbook of underwater imaging system design*. Tech. rep. Naval Undersea Center San Diego CA.

Garcia, R., T. Nicosevici, and X. Cufí (2002). "On the way to solve lighting problems in underwater imaging". In: *OCEANS'02 MTS/IEEE*. Vol. 2. IEEE, pp. 1018–1024.

Getreuer, P. (2012). "Automatic Color Enhancement (ACE) and its Fast Implementation Histogram Equalization". In: *Ipol* 2, pp. 1–12. ISSN: 2105-1232. DOI: `10.5201/ipol.2012.g-ace`.

Gevers, T., A. Gijsenij, J. Van de Weijer, and J.-M. Geusebroek (2012). *Color in computer vision: fundamentals and applications*. Vol. 23. John Wiley & Sons.

Ghani, A. S. A. and N. A. M. Isa (2017). "Automatic system for improving underwater image contrast and color through recursive adaptive histogram modification". In: *Computers and Electronics in Agriculture* 141, pp. 181 –195. ISSN: 0168-1699. DOI: `https://doi.org/10.1016/j.compag.2017.07.021`.

Godø, O. R., S. Johnsen, and T. Torkelsen (2014). "The LoVe Ocean Observatory is in Operation". In: *Marine Technology Society Journal* 48.2, pp. 24–30.

Gonzalez, R. C. and R. E. Woods (2008). *Digital image processing*. 3. ed. Upper Saddle River, NJ: Pearson Prentice Hall, p. 954. ISBN: 0-13-168728-X, 978-0-13-168728-8.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press.

Gorsky, G., M. D. Ohman, M. Picheral, S. Gasparini, L. Stemmann, J.-B. Romagnan, A. Cawood, S. Pesant, C. GarcÃŋa-Comas, and F. Prejger (2010). "Digital zooplankton image analysis using the ZooScan integrated system". In: *Journal of Plankton Research* 32.3, pp. 285–303. DOI: `10.1093/plankt/fbp124`.

Gouhier, T. C., A. Grinsted, and V. Simko (2016). *biwavelet: Conduct univariate and bivariate wavelet analyses*. (Version 0.20.10). URL: `http://github.com/tgouhier/biwavelet`.

Gracias, N., S. Negahdaripour, L. Neumann, R. Prados, and R. Garcia (Sept. 2008). "A motion compensated filtering approach to remove sunlight flicker in shallow water images". In: *OCEANS 2008*, pp. 1–7. DOI: `10.1109/OCEANS.2008.5152111`.

Hall-Spencer, J. M. (1998). "Conservation issues relating to maerl beds as habitats for molluscs". In: *Journal of Conchology* 2.2, pp. 271–286. ISSN: 00220019.

Harbitz, A. (2007). "Estimation of shrimp (Pandalus borealis) carapace length by image analysis". In: *ICES Journal of Marine Science* 64.5, pp. 939–944. ISSN: 10543139. DOI: `10.1093/icesjms/fsm047`.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning*. Vol. 18. Springer Series in Statistics 4. New York, NY: Springer New York, p. 746. ISBN: 978-0-387-84857-0. DOI: `10.1007/b94608`.

He, K., J. Sun, and X. Tang (2011). "Single image haze removal using dark channel prior". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.12, pp. 2341–2353. DOI: 10.1109/TPAMI.2010.168.

He, K., X. Zhang, S. Ren, and J. Sun (2016). "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of the IEEE International Conference on Computer Vision*. Vol. 11-18-December-2015, pp. 1026–1034. ISBN: 9781467383912. DOI: 10.1109/ICCV.2015.123. eprint: 1502.01852.

Henriques, M. C., R. Riosmena-Rodríguez, L. Coutinho, and M. Figueiredo (2014). "Lithophylloideae and Mastophoroideae (Corallinales, Rhodophyta) from the Brazilian continental shelf". In: *Phytotaxa* 190.1, pp. 112–129. ISSN: 1179-3163. DOI: 10.11646/phytotaxa.190.1.9.

Howe, B. M., F. K. Duennebier, and R. Lukas (2015). "The ALOHA cabled observatory". In: *SEAFLOOR OBSERVATORIES: A New Vision of the Earth from the Abyss*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 439–463. ISBN: 978-3-642-11374-1. DOI: 10.1007/978-3-642-11374-1_17.

Ingrid Kjerstad, I. N. (May 2016). "The effect of inherent optical properties of seawater on image quality". In: *Marine Imaging Workshop*. Southampton, UK. DOI: 10.13140/RG.2.1.4803.4161.

Ioccg (2000). "Remote Sensing of Ocean Colour in Coastal, and Other Optically-Complex Waters". In: *Reports of the International Ocean Colour Coordinating Group 3*, p. 144. ISBN: 3.

Iqbal, K., M. Odetayo, A. James, Rosalina Abdul Salam, and Abdullah Zawawi Hj Talib (Oct. 2010). "Enhancing the low quality images using Unsupervised Colour Correction Method". In: *2010 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, pp. 1703–1709. ISBN: 978-1-4244-6586-6. DOI: 10.1109/ICSMC.2010.5642311.

Jaffe, J. S. (Apr. 1990). "Computer modeling and the design of optimal underwater imaging systems". In: *IEEE Journal of Oceanic Engineering* 15.2, pp. 101–111. ISSN: 0364-9059. DOI: 10.1109/48.50695.

Jia, Y., E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell (2014). "Caffe: Convolutional Architecture for Fast Feature Embedding". In: *arXiv preprint arXiv:1408.5093*.

Johanson, A. N., S. Flögel, W.-C. Dullo, P. Linke, and W. Hasselbring (2017). "Modeling polyp activity of Paragorgia arborea using supervised learning". In: *Ecological Informatics* 39, pp. 109 –118. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2017.02.007.

Johnsen, G, Z Volent, E Sakshaug, F Sigemes, and L. H. Pettersson (2009). "6. Remote sensing in the Barents Sea". In: *Ecosystem Barents Sea*. Ed. by E. Sakshaug, G. H. Johnsen, and K. M. Kovacs. Tapir Academic Press, pp. 139–166. ISBN: 8251924618.

Johnsen, G., M. Ludvigsen, A. Sørensen, and L. M. S. Aas (2016). "The use of underwater hyperspectral imaging deployed on remotely operated vehicles - methods and applications". In: *IFAC-PapersOnLine* 49.23. 10th IFAC Conference on Control Applications in Marine SystemsCAMS 2016, pp. 476 –481. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2016.10.451.

Jonsson, L., P. Nilsson, F Floruta, and T Lundälv (2004). "Distributional patterns of macro- and megafauna associated with a reef of the cold-water coral Lophelia pertusa on the Swedish west coast". In: *Marine Ecology Progress Series* 284.15, pp. 163–171. ISSN: 01718630. DOI: 10.3354/meps284163.

Kaiser, M. J., ed. (2011). *Marine ecology.* eng. 2. ed. Oxford [u.a.]: Oxford Univ. Press, XVIII, 501 S. : zahlr. Ill., graph. Darst., Kt. ISBN: 978-0-19-922702-0, 0-19-922702-0.

Kempf, M. (1970). "Notes on the benthic bionomy of the N-NE Brazilian shelf". In: *Marine Biology* 5.3, pp. 213–224. ISSN: 00253162. DOI: 10.1007/BF00346909.

Kocak, D. M. and F. M. Caimi (2005a). "The Current Art of Underwater Imaging - With a Glimpse of the Past and Vision of the Future". In: *Marine Technology Society Journal* 39.3, pp. 5–26. ISSN: 00253324. DOI: 10.4031/002533205787442576.

Kocak, D. M. and F. M. Caimi (2005b). "The current art of underwater imaging–With a glimpse of the past and vision of the future". In: *Marine Technology Society Journal* 39.3, pp. 5–26.

Kocak, D. M., F. R. Dalgleish, F. M. Caimi, and Y. Y. Schechner (2008). "A focus on recent developments and trends in underwater imaging". In: *Marine Technology Society Journal* 42.1, pp. 52–67.

Kohonen, T. (1989). *Self-Organization and Associative Memory.* Vol. 8. Springer Series in Information Sciences. Berlin, Heidelberg: Springer Berlin Heidelberg, p. 312. ISBN: 978-3-540-51387-2. DOI: 10.1007/978-3-642-88163-3. arXiv: arXiv:1011.1669v3.

Lampitt, R., P. Favali, C. Barnes, M. Church, M. F. Cronin, K. Hill, Y. Kaneda, D. Karl, A. Knap, M. McPhaden, K. Nittis, I. Priede, J.-F. Rolin, U. Send, C.-C. Teng, T. Trull, D. Wallace, and R. Weller (Dec. 2010). "In Situ Sustained Eulerian Observatories". In: *Proceedings of OceanObs'09: Sustained Ocean Observations and Information for Society.* Venice, Italy: European Space Agency, 27pp. ISBN: 978-3-86987-200-1. DOI: 10.5270/OceanObs09.pp.27.

Langenkämper, D., M. Zurowietz, T. Schoening, and T. W. Nattkemper (2017). "BIIGLE 2.0 - Browsing and Annotating Large Marine Image Collections". In: *Frontiers in Marine Science* 4, p. 83. ISSN: 2296-7745. DOI: 10.3389/fmars.2017.00083.

Lecun, Y., L. Bottou, Y. Bengio, and P. Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. ISSN: 00189219. DOI: 10.1109/5.726791. arXiv: 1102.0183.

LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. DOI: 10.1109/5.726791.

Levinshtein, A., A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi (Dec. 2009). "TurboPixels: Fast Superpixels Using Geometric Flows". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.12, pp. 2290–2297. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2009.96.

Littler, M. M., D. S. Littler, S. M. Blair, and J. N. Norris (1986). "Deep-water plant communities from an uncharted seamount off San Salvador Island, Bahamas: distribution, abundance, and primary productivity". In: *Deep Sea Research Part A. Oceano-*

graphic Research Papers 33.7, pp. 881–892. ISSN: 01980149. DOI: `10.1016/0198-0149(86)90003-8`.

Littler, M. M., D. S. Littler, and M. D. Hanisak (1991). "Deep-water rhodolith distribution, productivity, and growth history at sites of formation and subsequent degradation". In: *Journal of Experimental Marine Biology and Ecology* 150.2, pp. 163–182. ISSN: 00220981. DOI: `10.1016/0022-0981(91)90066-6`.

Lu, H., Y. Li, Y. Zhang, M. Chen, S. Serikawa, and H. Kim (2017). "Underwater Optical Image Processing: a Comprehensive Review". In: *Mobile Networks and Applications* 22.6, pp. 1204–1211. ISSN: 1572-8153. DOI: `10.1007/s11036-017-0863-4`.

MacLeod, N., M. Benfield, and P. Culverhouse (Sept. 2010). "Time to automate identification". In: *Nature* 467.7312, pp. 154–155. ISSN: 0028-0836. DOI: `10.1038/467154a`.

MacQueen, J. et al. (1967). "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability.* Vol. 1. 14. Oakland, CA, USA., pp. 281–297.

Mahon, I., O. Pizarro, M. Johnson-Roberson, A. Friedman, S. B. Williams, and J. C. Henderson (May 2011). "Reconstructing pavlopetri: Mapping the world's oldest submerged town using stereo-vision". In: *2011 IEEE International Conference on Robotics and Automation*, pp. 2315–2321. DOI: `10.1109/ICRA.2011.5980536`.

Matabos, M., M. Hoeberechts, C. Doya, J. Aguzzi, J. Nephin, T. E. Reimchen, S. Leaver, R. M. Marx, A. Branzan Albu, R. Fier, U. Fernandez-Arcaya, and S. K. Juniper (2017). "Expert, Crowd, Students or Algorithm: who holds the key to deep-sea imagery 'big data' processing?" In: *Methods in Ecology and Evolution* 8.8, pp. 996–1004. ISSN: 2041-210X. DOI: `10.1111/2041-210X.12746`.

Maulik, U. and S. Bandyopadhyay (2002). "Performance evaluation of some clustering algorithms and validity indices". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.12, pp. 1650–1654.

Möller, T., I. Nilssen, and T. W. Nattkemper (Dec. 2016). "Data-Driven Long Term Change Analysis in Marine Observatory Image Streams". In: *2016 ICPR 2nd Workshop on Computer Vision for Analysis of Underwater Imagery (CVAUI)*. IEEE, pp. 13–18. ISBN: 978-1-5090-5870-9. DOI: `10.1109/CVAUI.2016.015`.

Möller, T., I. Nillsen, and T. W. Nattkemper (2017). "Active learning for the classification of species in underwater images from a fixed observatory". In: *Proceedings of The IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy.

Moore, A., J. Allman, and R. M. Goodman (Mar. 1991). "A real-time neural system for color constancy". In: *IEEE Transactions on Neural Networks* 2.2, pp. 237–247. ISSN: 1045-9227. DOI: `10.1109/72.80334`.

Morris, K. J., B. J. Bett, J. M. Durden, V. A. I. Huvenne, R. Milligan, D. O. B. Jones, S. McPhail, K. Robert, D. M. Bailey, and H. A. Ruhl. "A new method for ecological surveying of the abyss using autonomous underwater vehicle photography". In: *Limnology and Oceanography: Methods* 12.11 (), pp. 795–809. DOI: `10.4319/lom.2014.12.795`.

Mortensen, P. B. and J. H. Fosså (June 2006). "Species diversity and spatial distribution of invertebrates on deep-water Lophelia reefs in Norway". In: *Proceedings of 10th International Coral Reef Symposium*, pp. 1849–1868.

Nakamura, J. (2005). *Image Sensors and Signal Processing for Digital Still Cameras*, p. 350. ISBN: 0849335450. DOI: `10.1201/9781420026856`.

Nelson, W. (2009). "Calcified macroalgaeâĂŞcritical to coastal ecosystems and vulnerable to change: a review". In: *Marine and Freshwater Research* 60.8, pp. 787–801. ISSN: 1323-1650. DOI: `10.1071/MF08335`.

Nicosevici, T., N. Gracias, S. Negahdaripour, and R. Garcia (2009). "Efficient three-dimensional scene modeling and mosaicing". In: *Journal of Field Robotics* 26.10, pp. 759–788. ISSN: 1556-4967. DOI: `10.1002/rob.20305`.

Nilssen, I., Ø. Ødegård, A. J. Sørensen, G. Johnsen, M. A. Moline, and J. Berge (July 2015). "Integrated environmental mapping and monitoring, a methodological approach to optimise knowledge gathering and sampling strategy". In: *Marine Pollution Bulletin* 96.1-2, pp. 374–383. ISSN: 0025326X. DOI: `10.1016/j.marpolbul.2015.04.045`.

Nixon, M. and A. Aguado (2012a). "Chapter 1.4 - Computer vision systems". In: *Feature Extraction & Image Processing for Computer Vision, Third Edition*, pp. 12–15. ISBN: 978-0-12-396549-3. DOI: `http://dx.doi.org/10.1016/B978-0-12-396549-3.00001-X`.

Nixon, M. S. and A. S. Aguado (2012b). "Basic image processing operations". In: *Feature Extraction & Image Processing for Computer Vision*. Elsevier, pp. 83–136. ISBN: 0123725380. DOI: `10.1016/B978-0-12-396549-3.00003-3`.

Ocean Network Canada (2014). *NEPTUNE in the NE Pacific*. URL: `http://www.oceannetworks.ca/installations/observatories/neptune-ne-pacific/` (visited on 12/16/2015).

Ontrup, J and H Ritter (2002). "Hyperbolic Self-Organizing Maps for Semantic Navigation". In: *Advances in Neural Information Processing Systems 14* 14.14, pp. 1417–1424. ISSN: 10495258.

Ontrup, J., H. Wersing, and H. Ritter (Mar. 2004). "A computational feature binding model of human texture perception". In: *Cognitive Processing* 5.1, pp. 31–44. ISSN: 1612-4782. DOI: `10.1007/s10339-003-0003-x`.

Ontrup, J. and H. Ritter (July 2006). "Large-scale data exploration with the hierarchically growing hyperbolic SOM". In: *Neural Networks* 19.6-7, pp. 751–761. ISSN: 08936080. DOI: `10.1016/j.neunet.2006.05.015`.

Ontrup, J., N. Ehnert, M. Bergmann, and T. W. Nattkemper (May 2009). "BIIGLE-Web 2.0 enabled labelling and exploring of images from the Arctic deep-sea observatory HAUSGARTEN". In: *OCEANS 2009-EUROPE*. IEEE, pp. 1–7. DOI: `10.1109/OCEANSE.2009.5278332`.

Osterloff, J., I. Nilssen, M. A. de Oliveira Figueiredo, F. T. de Souza TÃćmega, T. MÃűller, and T. W. Nattkemper (May 2014a). "Automated Image based Biomass Quantification in Mesocosm Studies". In: *Geohab*. Lorne, Victoria, Australia. DOI: `10.13140/RG.2.2.28797.05602`.

Osterloff, J., T. Schoening, M. Bergmann, J. M. Durden, H. A. Ruhl, and T. W. Nattkemper (2014b). "Ranking Color Correction Algorithms using Cluster Indices". In: *Proceedigs of CVAUI 2014 (ICPR Workshop)*. Stockhom, Sweden. DOI: `10.1109/CVAUI.2014.13`.

Osterloff, J., I. Nilssen, and T. W. Nattkemper (2016a). "A computer vision approach for monitoring the spatial and temporal shrimp distribution at the LoVe observatory". In: *Methods in Oceanography* 15âĂŞ16. Computer Vision in Oceanography, pp. 114 –128. ISSN: 2211-1220. DOI: `10.1016/j.mio.2016.03.002`.

– (Sept. 2016b). "Computational Coral Feature Monitoring for the fixed underwater Observatory LoVe". In: *Proceedings of IEEE OCEANS 2016.* Monterey, CA, USA. DOI: `10.1109/OCEANS.2016.7761417`.

Osterloff, J., I. Nilssen, I. Eide, M. A. de Oliveira Figueiredo, F. T. de Souza Tâmega, and T. W. Nattkemper (June 2016c). "Computational Visual Stress Level Analysis of Calcareous Algae Exposed to Sedimentation". In: *PLoS ONE* 11.6, pp. 1–22. DOI: `10.1371/journal.pone.0157329`.

Osterloff, J., I. Nilssen, J. Järnegren, P. Buhl-Mortensen, and T. W. Nattkemper (2016d). "Polyp activity estimation and monitoring for cold water corals with a deep learning approach". In: *Proceedigs of CVAUI 2016 (ICPR Workshop).* Cancun, Mexico. DOI: `10.1109/CVAUI.2016.013`.

Osterloff, J., I. Nilssen, J. Järnegren, T. Van Engeland, P. Buhl-Mortensen, and T. W. Nattkemper (submitted July 2018). "Computer vision enables short- and long-term analysis of Lophelia pertusa polyp behaviour and colour from an underwater observatory". In: *Scientific Reports.*

Panetta, K., C. Gao, and S. Agaian (July 2016). "Human-Visual-System-Inspired Underwater Image Quality Measures". In: *IEEE Journal of Oceanic Engineering* 41.3, pp. 541–551. ISSN: 0364-9059. DOI: `10.1109/JOE.2015.2469915`.

Petit, F., P. Carre, and A. Capelle-Laize (Apr. 2009). "Underwater image enhancement by attenuation inversionwith quaternions". In: *Acoustics, Speech, and Signal Processing, IEEE International Conference on(ICASSP).* Vol. 00, pp. 1177–1180. DOI: `10.1109/ICASSP.2009.4959799`.

Pietikäinen, M., A. Hadid, G. Zhao, and T. Ahonen (2011). *Computer Vision Using Local Binary Patterns.* Vol. 40. Computational Imaging and Vision 11. London: Springer London, pp. 1–213. ISBN: 978-0-85729-747-1. DOI: `10.1007/978-0-85729-748-8`. arXiv: `arXiv:1011.1669v3`.

Pizer, S. M., E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld (1987). "Adaptive histogram equalization and its variations". In: *Computer Vision, Graphics, and Image Processing* 39.3, pp. 355–368. ISSN: 0734189X. DOI: `10.1016/S0734-189X(87)80186-X`.

Poynton, C. (2012). *Digital Video and HD: Algorithms and Interfaces.* The Morgan Kaufmann Series in Computer Graphics. Elsevier Science. ISBN: 9780123919328.

Price, D., A. J. Davies, and colleagues (Feb. 2017). "Time-lapse imaging reveals the ne-scale behaviour of Lophelia pertusa polyps in response to changing ow velocity and temperature". In: *Marine Imaging Workshop.* Kiel, Germany. URL: `http://marine-imaging-workshop.com/documents/miw17/presentations/private/MIW17_Poster_DavidPrice.pdf`.

Purser, a., J. Ontrup, T. Schoening, L. Thomsen, R. Tong, V. Unnithan, and T. W. Nattkemper (2013). "Microhabitat and shrimp abundance within a Norwegian cold-

water coral ecosystem". In: *Biogeosciences* 10.9, pp. 5779–5791. ISSN: 17264170. DOI: `10.5194/bg-10-5779-2013`.

Purser, A. (July 2015). "A Time Series Study of Lophelia pertusa and Reef Megafauna Responses to Drill Cuttings Exposure on the Norwegian Margin". In: *PLOS ONE* 10.7, pp. 1–29. DOI: `10.1371/journal.pone.0134076`.

Purser, A., M. Bergmann, T. Lundälv, J. Ontrup, and T. Nattkemper (Dec. 2009). "Use of machine-learning algorithms for the automated detection of cold-water coral habitats: a pilot study". In: *Marine Ecology Progress Series* 397, pp. 241–251. ISSN: 0171-8630. DOI: `10.3354/meps08154`.

Raitoharju, J., E. Riabchenko, K. Meissner, I. Ahmad, A. Iosifidis, M. Gabbouj, and S. Kiranyaz (Dec. 2016). "Data Enrichment in Fine-Grained Classification of Aquatic Macroinvertebrates". In: *2016 ICPR 2nd Workshop on Computer Vision for Analysis of Underwater Imagery (CVAUI)*, pp. 43–48. DOI: `10.1109/CVAUI.2016.020`.

Reinhard, E., M. Ashikhmin, B. Gooch, and P. Shirley (2001). "Color transfer between images". In: *IEEE Computer Graphics and Applications* 21.5, pp. 34–41. ISSN: 02721716. DOI: `10.1109/38.946629`.

Rensink, R. A., J. K. O'Regan, and J. J. Clark (2000). "On the Failure to Detect Changes in Scenes Across Brief Interruptions". In: *Visual Cognition* 7.1-3, pp. 127–145. DOI: `10.1080/135062800394720`.

Riosmena-Rodríguez, R, D. Steller, G Hinojosa-Arango, and M. Foster (2010). "Reefs that rock and roll: biology and conservation of rhodolith beds in the Gulf of California". In: *Marine Biodiversity and Conservation in the Gulf of California. University of Arizona Press and Sonoran Desert Museum, Tuscon, AZ*, pp. 49–71.

Rizzi, A., C. Gatta, and D. Marini (2002). "Color correction between gray world and white patch". In: *Electronic Imaging* 4662, pp. 367–375. ISSN: 0277786X. DOI: `10.1117/12.469534`.

– (2003). "A new algorithm for unsupervised global and local color correction". In: *Pattern Recognition Letters* 24.11, pp. 1663–1677. ISSN: 01678655. DOI: `10.1016/S0167-8655(02)00323-9`.

Roberts, J. M., A. Wheeler, A. Freiwald, and S. Cairns (2009). *Cold-Water Corals*. Vol. 23. Cambridge: Cambridge University Press, p. 226. DOI: `10.1017/CBO9780511581588`.

Roser, M., M. Dunbabin, and A. Geiger (May 2014). "Simultaneous underwater visibility assessment, enhancement and improved stereo". In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3840–3847. DOI: `10.1109/ICRA.2014.6907416`.

Ruderman, D. L., T. W. Cronin, and C.-C. Chiao (1998). "Statistics of cone responses to natural images: implications for visual coding". In: *Journal of the Optical Society of America A* 15.8, p. 2036. DOI: `10.1364/JOSAA.15.002036`.

Rumelhart, D. E., G. E. Hinton, R. J. Williams, et al. (1988). "Learning representations by back-propagating errors". In: *Cognitive modeling* 5.3, pp. 533–536.

Russakovsky, O., J. Deng, Z. Huang, A. C. Berg, and L. Fei-Fei (2013). "Detecting avocados to Zucchinis: What have we done, and where are we going?" In: *Proceed-*

ings of the IEEE International Conference on Computer Vision, pp. 2064–2071. ISBN: 9781479928392. DOI: `10.1109/ICCV.2013.258`.

Russakovsky, O., J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision* 115.3, pp. 211–252. ISSN: 15731405. DOI: `10.1007/s11263-015-0816-y`. eprint: `1409.0575`.

Rzhanov, Y, L. M. Linnett, and R Forbes (2000). "Underwater video mosaicing for seabed mapping". In: *Image Processing, 2000. Proceedings. 2000 International Conference on*. Vol. 1. IEEE, pp. 224–227. DOI: `10.1109/ICIP.2000.900935`.

Sanchez, G. (2012). *plsdepot: Partial Least Squares (PLS) Data Analysis Methods*. R package version 0.1.17. URL: `https://CRAN.R-project.org/package=plsdepot`.

Schanda, J. (July 2007). *Colorimetry*. Ed. by J. Schanda. Hoboken, NJ, USA: John Wiley & Sons, Inc., pp. 1–459. ISBN: 9780470175637. DOI: `10.1002/9780470175637`.

Schechner, Y. Y. and N. Karpel (2005). "Recovery of underwater visibility and structure by polarization analysis". In: *Oceanic Engineering, IEEE Journal of* 30.3, pp. 570–587.

Schettini, R. and S. Corchs (2010). "Underwater image processing: State of the art of restoration and image enhancement methods". In: *Eurasip Journal on Advances in Signal Processing* 2010. ISSN: 16876172. DOI: `10.1155/2010/746052`.

Schoening, T., N. Ehnert, J. Ontrup, and T. Nattkemper (July 2009). "BIIGLE Tools: A Web 2.0 Approach for Visual Bioimage Database Mining". In: *Information Visualisation, 2009 13th International Conference*, pp. 51–56. DOI: `10.1109/IV.2009.71`.

Schoening, T., D. Langenkämper, B. Steinbrink, D. Brün, and T. W. Nattkemper (Oct. 2015). "Rapid image processing and classification in underwater exploration using advanced high performance computing". In: *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–5. DOI: `10.23919/OCEANS.2015.7401952`.

Schoening, T., M. Bergmann, J. Ontrup, J. Taylor, J. Dannheim, J. Gutt, A. Purser, and T. W. Nattkemper (June 2012). "Semi-Automated Image Analysis for the Assessment of Megafaunal Densities at the Arctic Deep-Sea Observatory HAUSGARTEN". In: *PLoS ONE* 7.6, e38179. DOI: `10.1371/journal.pone.0038179`.

Schoening, T., T. Kuhn, D. O. Jones, E. Simon-Lledo, and T. W. Nattkemper (Apr. 2016a). "Fully automated image segmentation for benthic resource assessment of polymetallic nodules". In: *Methods in Oceanography* 15-16, pp. 78–89. ISSN: 22111220. DOI: `10.1016/j.mio.2016.04.002`.

Schoening, T., J. Osterloff, and T. W. Nattkemper (2016b). "RecoMIA - Recommendations for Marine Image Annotation: Lessons Learned and Future Directions". In: *Frontiers in Marine Science* 3.59. ISSN: 2296-7745. DOI: `10.3389/fmars.2016.00059`.

Shavlik, J. and T. Dietterich (1990). *Readings in Machine Learning*. Machine Learning Series. Morgan Kaufmann Publishers. ISBN: 9781558601437.

Shen-Chuan Tai Ting-Chou Tsai, J.-H. H. (2017). "Underwater image enhancement through depth estimation based on random forest". In: *Journal of Electronic Imaging* 26, pp. 26 –26 –11. DOI: `10.1117/1.JEI.26.6.063026`.

Shi, J. and J. Malik (Aug. 2000). "Normalized cuts and image segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.8, pp. 888–905. ISSN: 0162-8828. DOI: 10.1109/34.868688.

Siebeck, U. E., N. J. Marshall, a. Klüter, and O. Hoegh-Guldberg (2006). "Monitoring coral bleaching using a colour reference card". In: *Coral Reefs* 25.3, pp. 453–460. ISSN: 07224028. DOI: 10.1007/s00338-006-0123-8.

Simons, D. J. and R. A. Rensink (2005). "Change blindness: past, present, and future". In: *Trends in Cognitive Sciences* 9.1, pp. 16 –20. ISSN: 1364-6613. DOI: http://dx.doi.org/10.1016/j.tics.2004.11.006.

Singh, H, J Howland, and O Pizarro (2004). "Advances in large-area photomosaicking underwater". In: *IEEE Journal of Oceanic Engineering* 29.3, pp. 872–886. ISSN: 0364-9059 VO - 29. DOI: 10.1109/JOE.2004.831619.

Singh, H., C. Roman, O. Pizarro, R. Eustice, and A. Can (2007). "Towards high-resolution imaging from underwater vehicles". In: *The International journal of robotics research* 26.1, pp. 55–74.

Smith, A. R. (Aug. 1978). "Color Gamut Transform Pairs". In: *SIGGRAPH Comput. Graph.* 12.3, pp. 12–19. ISSN: 0097-8930. DOI: 10.1145/965139.807361.

Sokolova, M. and G. Lapalme (2009). "A systematic analysis of performance measures for classification tasks". In: *Information Processing & Management* 45.4, pp. 427 –437. ISSN: 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2009.03.002.

Sonka, M., V. Hlavac, and R. Boyle (2008). *Image Processing, Analysis and Machine Vision*. Vol. 4. Boston, MA: Springer US, p. 800. ISBN: 978-0-412-45570-4. DOI: 10.1007/978-1-4899-3216-7.

Steller, D. L., R. Riosmena-rodríguez, and M. S. Foster (2001). "6 Living Rhodolith Bed Ecosystems in the Gulf of California". In: *Ecosystems* 1944, pp. 72–82.

Stokes, M., M. Anderson, S. Chandrasekar, and R. Motta (1996). "A standard default color space for the internet-srgb". In: *Microsoft and Hewlett-Packard Joint Report*. Society for Imaging Science and Technology, pp. 1–17. URL: http://fp.optics.arizona.edu/opti588/reading/Stokes_sRGB_colorspace_detail.pdf.

Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich (June 2015). "Going deeper with convolutions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594.

Tai Sing Lee (1996). "Image representation using 2D Gabor wavelets". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18.10, pp. 959–971. ISSN: 01628828. DOI: 10.1109/34.541406.

Tâmega, F. T. S., P. S. Oliveira, and M. A. O. Figueiredo (2013). "Catalogue of the Benthic Marine Life for Peregrino Oil Field". In: *Campos Basin, Brazil. Instituto Biodiversidade Marinha, Rio de Janeiro*, p. 140.

Treibitz, T. and Y. Y. Schechner (Mar. 2009). "Active Polarization Descattering". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.3, pp. 385–399. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2008.85.

Trucco, E. and A. T. Olmos-Antillon (2006). "Self-tuning underwater image restoration". In: *Oceanic Engineering, IEEE Journal of* 31.2, pp. 511–519.

Turk, M. and A. Pentland (Jan. 1991). "Eigenfaces for Recognition". In: *Journal of Cognitive Neuroscience* 3.1, pp. 71–86. ISSN: 0898-929X. DOI: 10.1162/jocn.1991.3.1.71.

Tusa, E., A. Reynolds, D. Lane, N. Robertson, H. Villegas, and A. Bosnjak (Oct. 2014). "Implementation of a fast coral detector using a supervised machine learning and Gabor Wavelet feature descriptors". In: *Sensor Systems for a Changing Ocean (SSCO), 2014 IEEE*, pp. 1–6. DOI: 10.1109/SSCO.2014.7000371.

UNC Vision Lab (2016). *ILSVRC 2016*. URL: http://image-net.org/challenges/LSVRC/2016/results (visited on 03/13/2017).

– (2017). *ILSVRC 2017*. URL: http://image-net.org/challenges/LSVRC/2017/results (visited on 10/12/2017).

Vardaro, M. F., P. M. Bagley, D. M. Bailey, B. J. Bett, D. O. B. Jones, R. J. Milligan, I. G. Priede, C. M. Risien, G. T. Rowe, H. A. Ruhl, B. B. Sangolay, K. L. J. Smith, A Walls, and J Clarke (2013). "A Southeast Atlantic deep-ocean observatory: first experiences and results". In: *Limnology and Oceanography: Methods* 11, pp. 304–315. ISSN: 15415856. DOI: 10.4319/lom.2013.11.304.

Veksler, O., Y. Boykov, and P. Mehrani (2010). "Superpixels and Supervoxels in an Energy Optimization Framework". In: *Computer Vision – ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part V*. Ed. by K. Daniilidis, P. Maragos, and N. Paragios. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 211–224. ISBN: 978-3-642-15555-0. DOI: 10.1007/978-3-642-15555-0_16.

Witten, I. H., E. Frank, M. A. Hall, and C. J. Pal (2016). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wold, S, H Martens, and H Wold (2012). "The multivariate calibration problem in chemistry solved by the PLS method". In: *Matrix Pencils*. Vol. 973. Springer, pp. 286–293. ISBN: 978-3-540-11983-8. DOI: 10.1007/BFb0062108.

Wolfe, J. M., A. Reinecke, and P. Brawn (2006). "Why don't we see changes? The role of attentional bottlenecks and limited visual memory". In: *Visual Cognition* 14.4-8, pp. 749–780. DOI: 10.1080/13506280500195292.

Wynn, R. B., V. A. Huvenne, T. P. L. Bas, B. J. Murton, D. P. Connelly, B. J. Bett, H. A. Ruhl, K. J. Morris, J. Peakall, D. R. Parsons, E. J. Sumner, S. E. Darby, R. M. Dorrell, and J. E. Hunt (2014). "Autonomous Underwater Vehicles (AUVs): Their past, present and future contributions to the advancement of marine geoscience". In: *Marine Geology* 352.Supplement C. 50th Anniversary Special Issue, pp. 451 –468. ISSN: 0025-3227. DOI: 10.1016/j.margeo.2014.03.012.

Yang, M. and A. Sowmya (Dec. 2015). "An Underwater Color Image Quality Evaluation Metric". In: *IEEE Transactions on Image Processing* 24.12, pp. 6062–6071. ISSN: 1057-7149. DOI: 10.1109/TIP.2015.2491020.

Yuen, H., J Princen, J Illingworth, and J Kittler (1990). "Comparative study of Hough Transform methods for circle finding". In: *Image and Vision Computing* 8.1, pp. 71–77. ISSN: 02628856. DOI: 10.1016/0262-8856(90)90059-E.

Zhang, X., L. Hu, and M.-X. He (Mar. 2009). "Scattering by pure seawater: Effect of salinity". In: *Optics Express* 17.7, pp. 5698–5710. DOI: 10.1364/OE.17.005698.

Zibrowius, H. (1980). "Les Scléractiniaires de la Méditerranée et de l'Atlantique nord-oriental". In: *Mémoires de l'Institut océanographique, Monaco* 3. ISSN: 0304-5714.

Zuiderveld, K. (1994). "Contrast limited adaptive histogram equalization". In: *Graphics gems IV*. Academic Press Professional, Inc., pp. 474–485.