

AUTOMATED UNDERWATER OBJECT CLASSIFICATION USING OPTICAL IMAGERY

ASM Shihavuddin

Dipòsit legal: Gi. 701-2014
<http://hdl.handle.net/10803/132910>

ADVERTIMENT. L'accés als continguts d'aquesta tesi doctoral i la seva utilització ha de respectar els drets de la persona autora. Pot ser utilitzada per a consulta o estudi personal, així com en activitats o materials d'investigació i docència en els termes establerts a l'art. 32 del Text Refós de la Llei de Propietat Intel·lectual (RDL 1/1996). Per altres utilitzacions es requereix l'autorització prèvia i expressa de la persona autora. En qualsevol cas, en la utilització dels seus continguts caldrà indicar de forma clara el nom i cognoms de la persona autora i el títol de la tesi doctoral. No s'autoritza la seva reproducció o altres formes d'explotació efectuades amb finalitats de lucre ni la seva comunicació pública des d'un lloc aliè al servei TDX. Tampoc s'autoritza la presentació del seu contingut en una finestra o marc aliè a TDX (framing). Aquesta reserva de drets afecta tant als continguts de la tesi com als seus resums i índexs.

ADVERTENCIA. El acceso a los contenidos de esta tesis doctoral y su utilización debe respetar los derechos de la persona autora. Puede ser utilizada para consulta o estudio personal, así como en actividades o materiales de investigación y docencia en los términos establecidos en el art. 32 del Texto Refundido de la Ley de Propiedad Intelectual (RDL 1/1996). Para otros usos se requiere la autorización previa y expresa de la persona autora. En cualquier caso, en la utilización de sus contenidos se deberá indicar de forma clara el nombre y apellidos de la persona autora y el título de la tesis doctoral. No se autoriza su reproducción u otras formas de explotación efectuadas con fines lucrativos ni su comunicación pública desde un sitio ajeno al servicio TDR. Tampoco se autoriza la presentación de su contenido en una ventana o marco ajeno a TDR (framing). Esta reserva de derechos afecta tanto al contenido de la tesis como a sus resúmenes e índices.

WARNING. Access to the contents of this doctoral thesis and its use must respect the rights of the author. It can be used for reference or private study, as well as research and learning activities or materials in the terms established by the 32nd article of the Spanish Consolidated Copyright Act (RDL 1/1996). Express and previous authorization of the author is required for any other uses. In any case, when using its content, full name of the author and title of the thesis must be clearly indicated. Reproduction or other forms of for profit use or public communication from outside TDX service is not allowed. Presentation of its content in a window or frame external to TDX (framing) is not authorized either. These rights affect both the content of the thesis and its abstracts and indexes.



Universitat de Girona

PhD Thesis

Automated Underwater Object Classification using Optical Imagery

ASM Shihavuddin

2014



Universitat de Girona

PhD Thesis

Automated Underwater Object Classification using Optical Imagery

ASM Shihavuddin

2014

Doctoral Programme in Technology

Supervised by

Dr. Nuno Gracias

Dr. Rafael García

Work submitted to the University of Girona in partial fulfilment of the
requirements for the degree of Doctor of Philosophy

Publications

Within the scope of this thesis, the following publications were produced.

1. A. Shihavuddin, N. Gracias, R. Garcia, A. Gleason, and B. Gintert. Image-based coral reef classification and thematic mapping. *Remote Sensing*, 5(4) : 1809 - 1841, 2013. [96]
2. A. Shihavuddin, N. Gracias, R. Garcia, J. Escartin, and R. Pederson. Automated classification and thematic mapping of bacterial mats in the north sea. In *OCEANS'13 MTS/IEEE Conference*, Bergen, June 10-13, 2013. [95]
3. A. Shihavuddin, N. Gracias, and R. Garcia. Online sunflicker removal using dynamic texture prediction. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, pages 161-167. SciTePress, 2012. [94]
4. J. Escartin, R. Garcia, T. Barreyre, M. Cannat, N. Gracias, A. Shihavuddin and E. Mittelstaedt. Optical methods to monitor temporal changes at the seafloor: The Lucky Strike deep-sea hydrothermal vent field (Mid-Atlantic Ridge), *Underwater Technology Symposium (UT)*, 2013 IEEE International , vol.1(6), pp. 5-8, March, 2013. [22]

List of Acronyms

- AABB** Axis-Aligned Bounding Boxes
- AMI** Average Mutual Information
- ARMA** Auto-Regressive Moving Average process
- AUVs** Autonomous Underwater Vehicles
- CCA** Crustose Coralline Algae
- CIF** Cauchy's Integral Formula
- CLBP** Completed local binary pattern
- COTS** Crown Of Thorns starfish
- CURET** Columbia-Utrecht Reflectance and Texture
- DCT** Discrete Cosine Transforms
- DLBP** Dominant local binary pattern
- EMD** Earth Mover's Distance
- GIS** Geographic Information Systems
- GLCM** Grey Level Co-occurrence Matrix
- GPGPU** General Purpose Graphics Processing Units
- HSV** Hue Saturation Value
- IID** Independent and Identically Distributed process
- JPEG** Joint Photographic Experts Group
- KNN** K Nearest Neighbor
- KPCA** Kernel Principal Component Analysis
- KTH-TIPS** Kungliga Tekniska Hogskolan Textures under varying Illumination, Pose and Scale
- LBP** Local Binary Patterns
- LDA** Linear Discriminant Analysis
- LHC** Local Homogeneity Coefficient
- LIBSVM** LIBrary for Support Vector Machines
- MCR-LTER** Moorea Coral Reef Long Term Ecological Research site
- MLC** Moorea Labeled Corals
- MR** Maximum Response
- NOAA** National Oceanic and Atmospheric Administration
- NCC** Normalized Chromaticity Coordinates
- NN** Neural Network

OA Overall Accuracy
OLLDS Open Loop Linear Dynamic System
PCA Principal Component Analysis
PDWMD Probability Density Weighted Mean Distance
PTLSC Probabilistic Targets Least Square Classifier
ROVs Remotely Operated Vehicles
RSMAS Rosenstiel School of Marine and Atmospheric Sciences
SIFT Scale Invariant Feature Transform
SfM Structure from Motion
SVM Support Vector Machine
UIUC University of Illinois at Urbana-Champaign

List of Figures

| | | |
|------|--|----|
| 2.1 | General framework for supervised object classification | 10 |
| 3.1 | Refracted sunlight creating illumination patterns on the sea floor, varying in space & time following the dynamics of surface waves. | 26 |
| 3.2 | Example of rapid changes in the appearance of underwater objects in consecutive images acquired in a short time interval. | 26 |
| 3.3 | Effects of sunflicker removal in a mosaic creation. | 27 |
| 3.4 | Projected texture mapping on the <i>reference</i> images from different neighboring images | 32 |
| 3.5 | Example of a created 3D-based median image. | 32 |
| 3.6 | Step-by-step flow diagram of the proposed online method for sunflicker removal. | 34 |
| 3.7 | Illustration of \mathbf{W}_{t-1} matrix construction using the difference images of the <i>Grounding</i> dataset. | 35 |
| 3.8 | Step by step results of online Sunflicker removal for the <i>Grounding</i> sequence. | 37 |
| 3.9 | Examples of single frames from the <i>Grounding</i> , <i>Rocky</i> , <i>Andros</i> and <i>uneven</i> sequences | 38 |
| 3.10 | Sunflicker pattern extracted from a swimming pool video. | 40 |
| 3.11 | Comparison between the proposed and the offline method [37] for the <i>Grounding</i> sequence | 41 |
| 3.12 | Illustration of the online sunflicker removal performance | 42 |
| 3.13 | Performance comparison with the <i>Grounding</i> video sequence | 43 |
| 3.14 | Comparison between the proposed and the offline method for the <i>Rocky</i> and <i>Andora</i> sequences | 44 |

LIST OF FIGURES

| | | |
|------|---|----|
| 3.15 | Performance comparison in the <i>Synthetic</i> sequences between the on-line and offline approaches using homography median in both cases. | 44 |
| 3.16 | Performance comparison graph on the <i>Uneven</i> sequence | 45 |
| 4.1 | Proposed framework for supervised object classification using underwater optical imagery. | 48 |
| 4.2 | Illustration of the presence of color markers in raw images in the Moorea Coral Reef (MCR) dataset and color corrected images. | 51 |
| 4.3 | Output of CLAHS for different numbers of subregions. | 53 |
| 4.4 | Example of the color stretching. | 53 |
| 4.5 | An example of superpixels segmentation using the TurboPixels method. | 55 |
| 4.6 | Effects on classification accuracy by varying the number of superpixels. | 56 |
| 4.7 | Examples of Gabor filter wavelet with scale variation. | 57 |
| 4.8 | Examples of Gabor filter wavelet with orientation variation. | 58 |
| 4.9 | Flowchart of Completed Local Binary Pattern (CLBP) feature generation. | 61 |
| 4.10 | Example of a Completed Local Binary Pattern (CLBP) creation with a center pixel; its P circularly & evenly spaced neighbors with radius R | 61 |
| 4.11 | Example of Completed Local Binary Pattern (CLBP) generation. | 62 |
| 4.12 | 3D reconstruction of a portion of the <i>Ordnance</i> dataset | 65 |
| 4.13 | Step by step illustration of the process of creating elevation map. | 66 |
| 4.14 | Example of mosaic image with corresponding elevation map created from <i>Ordnance dataset</i> | 67 |
| 4.15 | Example of polynomial surface fitting of a patch from <i>Ordnance dataset</i> using sparse points. | 68 |
| 4.16 | Example of surface area (A_s) and area on the principal plane (A_p) of the patch. | 70 |
| 4.17 | Examples of phase symmetry images from corresponding elevation maps containing bombshells and background respectively. | 71 |
| 4.18 | Example of variations in histogram bin counts after applying Chi-square kernel mapping. | 73 |
| 4.19 | Flowchart of the implementation of the proposed method. | 79 |
| 5.1 | Precision-recall curves for individual classes of the <i>MLC 2008</i> dataset using our classification method. | 91 |

LIST OF FIGURES

| | | |
|------|--|-----|
| 5.2 | The time required to classify the <i>RSMAS</i> dataset using four test methods. | 96 |
| 5.3 | The overall accuracy of each method as a function of the number of images in the training data. This test was done on the <i>MLC 2008</i> dataset. | 97 |
| 5.4 | Effects of morphological filtering on the classification results. | 98 |
| 5.5 | Thematic mapping of the <i>Red Sea mosaic</i> using our proposed method. | 99 |
| 5.6 | The accuracy of the tested classification methods applied to the <i>Red Sea mosaic</i> | 100 |
| 5.7 | Location of misclassifications in thematic mapping by different methods shown in red in the <i>Red Sea mosaic</i> | 101 |
| 5.8 | Comparison of thematic mapping results for the <i>North Sea</i> mosaic. | 102 |
| 5.9 | Thematic mapping created on the <i>Grounding</i> mosaic without sun-flicker removal | 104 |
| 5.10 | Thematic mapping created on <i>Grounding mosaic</i> with sunflicker removal | 105 |
| 5.11 | Thematic mapping with <i>Ordnance</i> dataset using 2D, 2.5D and 2.5D + 3D features. | 107 |
| 5.12 | Precision-recall curves for both background and bombshells on the validation set of the <i>Ordnance</i> dataset. | 108 |
| 5.13 | Thematic mapping performed on the <i>Ordnance</i> mosaic with the use of both 2D and 2.5D features. | 109 |
| 5.14 | Thematic mapping using <i>Grounding mosaic</i> of the year 2005. | 112 |
| 5.15 | Thematic mapping using <i>Grounding mosaic</i> of the year 2006. | 113 |
| 5.16 | Thematic mapping using <i>Grounding mosaic</i> of the year 2011. | 114 |
| 5.17 | Thematic mapping using <i>Grounding mosaic</i> of the year 2012. | 115 |
| A.1 | Example of image patches from the <i>EILAT</i> dataset. | 125 |
| A.2 | A subset of the <i>RSMAS</i> dataset. | 126 |
| A.3 | A subset of the <i>2008</i> dataset showing two examples for each class. | 127 |
| A.4 | A subset of the <i>EILAT 2</i> dataset showing 4 examples of each class. | 128 |
| A.5 | Examples of training patches of each individual class (shell chaff, bacterial mat, sand) of the <i>BMAT</i> dataset. | 129 |
| A.6 | A subset of <i>UIUCtex</i> dataset showing 4 examples from each of 5 classes. | 130 |

LIST OF FIGURES

| | | |
|------|---|-----|
| A.7 | A subset of the <i>CURET</i> texture dataset showing 4 examples from each of the three classes. | 130 |
| A.8 | A subset of the <i>KTH-TIPS</i> dataset showing six examples from 2 of the 10 classes. | 131 |
| A.9 | The <i>Red Sea mosaic</i> image covering an area of 19.2 square meters. . . | 132 |
| A.10 | The <i>North Sea mosaic</i> image covering an area of 232.86 square meters | 134 |
| A.11 | The <i>Grounding mosaic</i> image covering an area of 291 square meters . | 135 |
| A.12 | The <i>Ordnance mosaic</i> image having the resolution of $7,949 \times 8,444$ pixels. | 137 |
| A.13 | Examples of small parts of the <i>Ordnance</i> mosaic. | 138 |

List of Tables

| | | |
|-----|--|----|
| 2.1 | A brief summary of methods classifying benthic images. | 14 |
| 2.2 | A numerical definition of a confusion matrix | 21 |
| 3.1 | Breakdown of the average running time required for each step in the proposed online sunflicker removal method. | 45 |
| 4.1 | The list of indicators used for computing the co-occurrence matrix based features. | 59 |
| 4.2 | The list of statistics used as GLCM features. | 60 |
| 5.1 | A brief summary of the characteristics of the underwater image datasets used in this work for classification and thematic mapping. | 83 |
| 5.2 | Percentage of overall accuracy with different image enhancement options as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. | 85 |
| 5.3 | Percentage of overall accuracy with different feature extraction methods as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. | 86 |
| 5.4 | Percentage of overall accuracy with different normalization and kernel mapping methods, as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. | 87 |
| 5.5 | Percentage of overall accuracy with different dimension reduction and classification methods. | 88 |
| 5.6 | Selected configuration of our proposed method for all the datasets used in experimentation. | 89 |
| 5.7 | The error matrix of our proposed classification method tested on the <i>MLC 2008</i> dataset. | 92 |

LIST OF TABLES

| | | |
|------|---|-----|
| 5.8 | The error matrix of the Marcos method [65] tested on the <i>MLC 2008</i> dataset. | 93 |
| 5.9 | The error matrix of the Stokes & Deane [100] method tested on the <i>MLC 2008</i> dataset. | 93 |
| 5.10 | The error matrix of the Pizarro method [80] tested on the <i>MLC 2008</i> dataset. | 94 |
| 5.11 | The error matrix of the Beijbom method [2] tested on the <i>MLC 2008</i> dataset. | 94 |
| 5.12 | Comparison based on overall accuracy (OA) among each method . . . | 94 |
| 5.13 | Comparison based on Average precision (AP) among each method . . | 95 |
| 5.14 | Comparison between a) Classifying the mosaic image directly and b) Creating a mosaic from the classified images. | 98 |
| 5.15 | Summary and comparison of classification results using 2.5D features with <i>Ordnance</i> dataset. | 108 |
| 5.16 | Classification and thematic mapping result for <i>Grounding mosaic</i> created over the years of 2005, 2006, 2011 and 2012. | 110 |
| A.1 | A brief summary of the underwater image datasets used in this work for classification and thematic mapping. | 124 |

Acknowledgements

This work is partially funded by the MICINN under the FI program, the Spanish MICINN under grant CTM2010-15216 (MuMap) and by the EU Project FP7-ICT-2011-288704 (MORPH) and US DoD/DoE/EPA project ESTCP SI2010. The author would like to acknowledge Assaf Zvuloni and Yossi Loya of the Tel Aviv University for the use of the Red Sea survey imagery. We would also like to thank Kasey Cantwell and R. Pamela Reids group at the University of Miami for help on the creation of the EILAT and RSMAS datasets, Oscar Beijbom for making his implementation publicly available on the web and also the authors of Weka, CLBP, Prtools and Vlfeat toolboxes.

All the credit goes to the almighty god who made me able to come this far. I would like to thank my supervisor Dr. Nuno Gracias and Dr. Rafael Garcia for providing me with all the guidance, encouragements and also for having the faith in me. Also I would like to show my gratitude for my family, my friends from the VIBOT generations who had been as my family and as my friends during all this time of during my PhD. Also I would like to regard all the members of VICOROB group who had been extremely supportive and motivating for finishing this work.

Contents

| | |
|---|-------------|
| List of Figures | v |
| List of Tables | ix |
| Contents | xiii |
| 1 Introduction | 1 |
| 1.1 Background | 1 |
| 1.2 Objectives | 2 |
| 1.3 Motivation | 3 |
| 1.4 Challenges | 4 |
| 1.5 Contributions | 5 |
| 1.6 Outline of the thesis | 6 |
| 2 Literature review | 9 |
| 2.1 General framework of object classification | 9 |
| 2.2 Related works on automated benthic habitat classification | 10 |
| 2.3 Image based classification | 16 |
| 2.3.1 Pizarro | 16 |
| 2.4 Patch based classification | 17 |
| 2.4.1 Marcos | 17 |
| 2.4.2 Stokes & Deane | 18 |
| 2.4.3 Beijbom | 19 |
| 2.5 Accuracy assessment methods | 19 |
| 2.6 Summary | 24 |
| 3 Online Sunflicker Removal Using Dynamic Texture Prediction | 25 |

CONTENTS

| | | |
|----------|---|-----------|
| 3.1 | Background | 25 |
| 3.2 | Related work | 27 |
| 3.3 | Framework | 28 |
| 3.3.1 | Dynamic texture modeling and synthesizing | 29 |
| 3.3.2 | Motion compensated filtering | 30 |
| 3.3.3 | Median image using 3D registration | 31 |
| 3.4 | Proposed algorithm | 33 |
| 3.5 | Sunflicker datasets | 37 |
| 3.6 | Methodology | 40 |
| 3.7 | Results | 41 |
| 3.8 | Summary | 46 |
| 4 | Proposed configurable classification framework | 47 |
| 4.1 | Introduction | 47 |
| 4.2 | Overview of related methods | 49 |
| 4.2.1 | Pre-processing | 49 |
| 4.2.2 | Segmentation | 54 |
| 4.2.3 | Feature extraction | 55 |
| 4.2.4 | Feature modification | 72 |
| 4.2.5 | Learning | 74 |
| 4.2.6 | Classification | 76 |
| 4.3 | Implementation | 78 |
| 4.4 | Summary | 78 |
| 5 | Results and Analysis | 81 |
| 5.1 | Methodology for evaluation | 81 |
| 5.1.1 | Classification performance evaluation | 81 |
| 5.1.2 | Configuration selection | 82 |
| 5.1.3 | Dataset characteristics | 83 |
| 5.1.4 | Thematic mapping accuracy measurement | 83 |
| 5.2 | Experimentation with different configurations | 84 |
| 5.2.1 | Pre-processing sub-steps | 84 |
| 5.2.2 | Feature extraction sub-steps | 85 |
| 5.2.3 | Feature modification and learning | 86 |
| 5.2.4 | Selected configurations and recommendations for future datasets | 87 |

| | | |
|----------|---|------------|
| 5.3 | Evaluation of the proposed method | 91 |
| 5.4 | Comparison with other methods | 92 |
| 5.5 | Classifying mosaic image | 96 |
| 5.5.1 | Thematic mapping directly on mosaic image | 97 |
| 5.5.2 | Analysis on sunflicker removed mosaic | 103 |
| 5.5.3 | Impact of 3D features | 106 |
| 5.5.4 | Quantifying change detection | 108 |
| 5.6 | Closing remarks | 110 |
| 6 | Conclusions | 117 |
| 6.1 | Summary | 117 |
| 6.2 | Main contributions | 119 |
| 6.3 | Limitations | 120 |
| 6.4 | Future work | 120 |
| A | Datasets | 123 |
| A.1 | Benthic datasets | 123 |
| A.1.1 | <i>EILAT</i> | 125 |
| A.1.2 | <i>RSMAS</i> | 125 |
| A.1.3 | <i>MLC 2008</i> | 127 |
| A.1.4 | <i>EILAT 2</i> | 128 |
| A.1.5 | <i>BMAT</i> | 128 |
| A.2 | Texture datasets | 129 |
| A.2.1 | <i>UIUCtex</i> | 129 |
| A.2.2 | <i>CURET</i> | 130 |
| A.2.3 | <i>KTH-TIPS</i> | 130 |
| A.3 | Mosaic datasets | 131 |
| A.3.1 | The <i>Red Sea mosaic</i> | 131 |
| A.3.2 | The <i>North Sea mosaic</i> | 133 |
| A.3.3 | The <i>Grounding mosaic</i> | 133 |
| A.4 | <i>Ordnance dataset</i> | 136 |
| A.5 | Summary | 136 |
| | Appendices | 122 |
| | References | 139 |

CONTENTS

Resum

Aquesta tesi tracta el problema de la caracterització automàtica d'imatges òptiques submarines. L'adquisició automatitzada d'imatges submarines permet l'obtenció i emmagatzematge de grans quantitats de dades per les quals la classificació manual pot requerir mesos de feina. Una classificació automatitzada i supervisada d'aquestes dades suposa un estalvi significatiu de temps i recursos, així com permet l'extracció d'informació valuosa per a realitzar estudis marins i geològics. Pocs són els treballs en la bibliografia orientats a la resolució d'aquest problema. A més, aquests pocs treballs no són capaços d'obtenir resultats consistents quan s'apliquen sobre una mostra variada d'imatges òptiques subaquàtiques.

En aquest treball es presenta un nou sistema de classificació d'imatges òptiques subaquàtiques, que pot ser aplicat tant a imatges individuals com a mosaics d'imatges. El mètode proposat es pot configurar d'acord amb les característiques individuals de les dades, com ara la mida del conjunt de dades, el nombre de classes, la resolució de les mostres, la disponibilitat d'informació de color i els diferents tipus de classes. El document presenta la combinació de característiques i classificadors que aconseguen els millors resultats sobre imatges subaquàtiques, així com les directrius per a la seva selecció. A més es presenta una nova solució per a l'eliminació de càustiques a les imatges subaquàtiques de poca profunditat, que s'introdueix com una etapa de preprocessat prèvia al sistema de classificació proposat.

També s'han examinat els mètodes principals de l'estat de l'art per a la classificació d'imatges submarines i els més rellevants han estat comparats amb el mètode proposat, utilitzant un mateix conjunt d'imatges bentòniques. El mètode proposat aconseguix la major precisió de classificació de manera general respecte als altres mètodes de l'estat de l'art, mantenint un temps d'execució moderat.

El sistema proposat també s'ha aplicat sobre mosaics d'imatges de grans dimensions del Mar Roig i el Mar del Nord amb l'objectiu de crear mapes temàtics completament classificats del fons marí.

Resumen

Esta tesis trata el problema de la caracterización automática de imágenes óptica submarinas. La adquisición automatizada de imágenes submarinas permite la obtención y almacenamiento de grandes cantidades de datos cuya clasificación manual puede requerir meses de trabajo. Una clasificación automatizada y supervisada de dichos datos supone un ahorro significativo de tiempo y recursos, así como permite la extracción de información valiosa para realizar estudios marinos y geológicos. Pocos son los trabajos en la bibliografía orientados a la resolución de dicho problema. Además, esos pocos trabajos no son capaces de obtener resultados consistentes cuando se aplican sobre una muestra variada de imágenes ópticas subacuáticas.

En este trabajo se presenta un novedoso sistema de clasificación de imágenes ópticas subacuáticas, que puede ser aplicado tanto a imágenes individuales como a mosaicos de imágenes. El método propuesto puede ser configurado de acuerdo con las características individuales de los datos, tales como el tamaño del conjunto de datos, el número de clases, la resolución de las muestras, la disponibilidad de información de color y los diferentes tipos de clases. El documento presenta la combinación de características y clasificadores que logran los mejores resultados sobre imágenes subacuáticas, así como las directrices para su selección. Además se presenta una novedosa solución para la eliminación de cáusticas en las imágenes subacuáticas de poca profundidad, que se introduce como una etapa de preprocesado previa al sistema de clasificación propuesto.

También se han examinado los métodos principales del estado del arte para la clasificación imágenes submarinas y los más relevantes han sido comparados con el método propuesto, utilizando un mismo conjunto de imágenes bentónicas. El método propuesto logra la mayor precisión de clasificación de forma general respecto a los otros métodos del estado del

arte, manteniendo un tiempo de ejecución moderado.

El sistema propuesto también se ha aplicado sobre mosaicos de imágenes de grandes dimensiones del Mar Rojo y el Mar del Norte con el objetivo de crear mapas temáticos completamente clasificados del fondo marino.

Abstract

This thesis addresses the problem of automated underwater optical image characterization. Remote underwater optical sensing allows the collection and storage of vast amounts of data for which manual classification may take months. Supervised automated classification of such datasets can save time and resources and can also enable extraction of valuable information related to marine and geological research.

There are few works in the literature that have addressed this problem, and these few fail to perform consistently well on varied underwater optical image datasets.

This work proposes a novel image classification framework for underwater optical images applicable to both single images and composite mosaic datasets. The proposed method can be configured to the characteristics of individual datasets such as the size of the dataset, number of classes, resolution of the samples, color information availability and class types. The combination of features and classifiers that attain the best results for underwater images is also presented in this thesis together with the guidelines for selection. Additionally we propose a novel approach for online sunflicker removal, which turns out to be an important pre-processing sub-step in the proposed framework for classification of shallow water imaging.

In this study, a review of the main underwater image classification methods is presented and the most relevant ones are compared with our proposed method on a common benthic datasets. The proposed method achieves the highest overall classification accuracy when compared with the state of the art methods and requires moderate execution time.

Our developed framework is also applied to large-scale image mosaics of the Red Sea and the North Sea to create completely classified thematic

maps of the seabed.

Chapter 1

Introduction

An exciting underwater world is waiting to be explored using the technological advances achieved in recent years. Several research groups all over the world are currently working in the field of underwater imaging and mapping. In this thesis, we develop a tool for automatic analysis of underwater optical imagery where the final product is a thematic map encoding the seabed characterization. Related policy makers, scientists, and researchers may use these classified maps as a tool for a better understanding of the subject.

1.1 Background

Remote sensing technologies such as acoustic mapping, multi-beam sonar and optical imaging among many others, are being increasingly applied for seabed observation. In this work, we focus on optical imagery, one of the remote sensing technologies that is commonly used for seafloor exploration, archeology, marine geology and marine biology and biodiversity [104], among other fields.

Underwater imaging technologies have improved considerably with the recent advancements in digital consumer cameras. Moreover, with the help of Remotely Operated Vehicles (ROVs) and Autonomous Underwater Vehicles (AUVs), thousands of images are now easily collectable. These possibilities increase the need of an automated analysis system of the underwater images to be used by human analysts. Using only manual analysis, acquiring the potential benefit of such large

1. INTRODUCTION

datasets is nearly impossible.

Only few projects in the past have addressed this problem. However, their main drawbacks are the inconsistent accuracy on different challenging datasets and the use of a fixed framework for all types of datasets. In consequence, they can hardly be considered a reliable tool for automated analysis of underwater images. Moreover, the methods were never compared on common datasets to derive constructive conclusions.

Recently, many methods for real world object classification have been developed that work very well on structured objects and textures, given a sufficient number of training sets. These methods are yet to be tested in an underwater environment where optical imaging comes with unique challenges. Effects such as blurring, scattering, sunflicker and color attenuation, among many others, need to be addressed before the desired classification accuracy can be achieved.

1.2 Objectives

Supervised automated annotation tools for underwater optical images have become necessary for analysing large amounts of acquired datasets. The main goal in this work is to explore the possibility of developing such classification methods for underwater optical images that finally result in a thematic map of the seabed in the surveyed area. Achieving reliable classification accuracy in underwater images is a difficult problem. The main difficulties are the underwater artifacts in optical images, the lack of training data, significant intra-class and inter-site variability and complex spatial borders between classes, among others.

In this work, we aim to evaluate the available state of the art computer vision techniques for classifying underwater objects with sufficient accuracy. We also aim to generate methods to remove unwanted artifacts from underwater optical images with specific attention to sunflicker effects. As mentioned above, usefulness of the existing methods for underwater image classification has not been demonstrated on common datasets. Without a comparison using common datasets, it is impossible to assess the relative effectiveness and efficiency of these methods. One aim of this thesis is to compare the existing most prominent methods together with our proposed

one, on common datasets in order to have a relevant comparative assessment. We would like to provide a complete review of the state-of-the-art underwater object classification using optical imagery and find the approaches that provide possibilities for use in future investigation.

We would also like to develop a classification framework that works consistently well on various underwater optical datasets and is hardware efficient and application oriented. For different marine and geological applications, such a classification tool should be very useful for experts in the field. Finally, we would like to ensure that the proposed method requires the least amount of user input as possible.

1.3 Motivation

Underwater image mapping can have a wide variety of useful applications in the field of coastal management, ecological analysis and geological exploration. For example, underwater ecosystems provide a reliable indication of the impact of global temperature changes. With the rapid growth of urbanization and the change in human life styles, there have been huge increases in shipping, port and industrial activities, water pollution, exploration for oil and gas and recreational uses such as boating and fishing. In order to evaluate the impact of these activities, regular monitoring of such events is very important. Proper policies adapted in order to analyze and curtail these impacts can greatly benefit global environmental sustainability.

We focus on remote sensing using digital underwater imagery as it contains several useful features for more accurate seabed characterization. First, digital images have an outstanding archival potential. They can be easily stored, shared, and analyzed for different purposes, by different people, or using multiple methods. Second, imagery can be collected from a variety of platforms, such as AUVs, ROVs, towed arrays or drop cameras, thereby allowing data to be collected in places or over temporal and spatial scales that a traditional dive survey simply could not accomplish [1].

Underwater image classification is a relatively new field of research with unique challenges but with great future potential. It is very exciting to work in this area of research, as it is closely related with global well being, with a greater possibility of

1. INTRODUCTION

contributing and acquiring visually comprehensible results. Many research projects addressing this important issue are currently being carried out. In the near future, automated underwater image classification will be very essential for an accurate analysis of the varied survey data.

In general, building an automated system that can reduce human labor is a challenging task. To keep up with the pace of exponentially increased use of digital cameras and videos, the need for automated analysis of such data is evident. Many works have focused on the real-world environment, whereas a big gap remains in the underwater imagery field. We have dedicated our work in this thesis to address such issues and find a feasible solution that can be considered as state of the art in this field.

1.4 Challenges

Underwater image classification is still a relatively new area of research compared with the existing large body of work from computer vision research on man-made life objects. Underwater imagery comes together with several unique challenges that need to be specifically addressed.

A few of the most challenging obstacles to classification accuracy of underwater images include:

- significant intra-class and inter-site variability in the morphology of seabed organisms [2] or structures of interest,
- complex spatial borders between classes,
- subjective annotation of training data by different analysts,
- variations in viewpoint, distance, and image quality,
- limits to spatial and spectral resolution when trying to classify to a free taxonomic scale,
- partial occlusion of objects due to the three-dimensional structure of the seabed,
- gradual changes in the structures of the classes,

-
- lighting artifacts due to wave focusing [37; 89; 94] and
 - variable optical properties of the water column.

All these individual challenges or limitations are subject to detailed research by the computer vision community. In this thesis, we make use of available state-of-the-art methods to mitigate some of them to a satisfactory level. However, not all of them have adequate solutions available in the literature. For lighting artifacts, we proposed a novel method for online sunflicker removal as discussed in Chapter 3.

Together with such challenges, on the one hand, there are many classes underwater that share samples in the same class but with significant differences between each other in terms of shape, color, texture, size, rotation, illumination, view angle, camera distance and light conditions. On the other hand, there are overlapping classes that look almost the same from specific angles and distances. When objects are classified manually given a single point, they get labeled using information from the perspective, neighbors and surroundings if needed. In a small patch, this surrounding information is often not available. Therefore, positioning and scaling of the patch size is an open question. When creating training sets, manual annotation can also be subject to variations from person to person. Even the same person may classify the same examples differently in different runs [16]. In addition, there can be many previously unknown classes present in underwater images which might not have enough training examples to allow adequate class learning. The datasets used in this work are described in details in Appendix A, which illustrates the challenges mentioned above.

1.5 Contributions

The main contribution of the work in this thesis can be summarized as follows

- We have developed a novel framework for under underwater optical image classification that can perform consistently well on different challenging seabed datasets. The proposed method is able to classify both on individual images or directly on composite mosaics. When compared with several challenging datasets, our proposed method produced consistently better classification accuracy than other available state of the art methods. As a final product, the

1. INTRODUCTION

proposed system generates thematic maps of the sea bed. In our work, we define *thematic mapping* as the color coded segmentation of a map into several different classes that keep a spatial coherence in order to obtain a diagrammatic representation of an area on the seafloor.

- We present a detailed review of the state-of-the-art methods for underwater object classification with optical imagery and a comparative analysis of the state-of-the-art methods on common datasets. Previously, all of these works were only compared with their individual datasets mentioned in the respective papers.
- Our proposed method is configurable with the characteristics of the dataset. With this adaptable configuration, it is possible to tune the system to get sufficiently good accuracy for any new dataset in the first go. In this thesis, we analyze the characteristics of the datasets with available methods in several steps of the proposed framework and came up with a guideline to make this appropriate selection.
- In the preprocessing step of the classification framework, we needed to remove sunflicker artifacts from the image whenever strong refracted sunlight was present. In this work, we propose a novel online sunflicker removal method using dynamic texture prediction and temporal or 3D based median image. This new approach produces better results than the state-of-the-art methods, even on image sets with very strong spatio-temporal sunflicker patterns.
- We also investigate the use of 3D structure information to obtain features that improve the classification.

Main contributions of this work have been published in one journal [96] and two conferences [94; 95].

1.6 Outline of the thesis

This thesis contains sequentially a literature review, a novel sunflicker removal method, the proposed classification framework, and finally the results and conclusions chapters. A description of the datasets used for testing are given in Appendix A.

Chapter 2 describes the related state-of-the-art works on underwater object classification using optical imagery. In the later part, it also describes the related theory of the classification accuracy assessment methods used.

Chapter 3 presents the novel idea of online sunflicker method using dynamic texture prediction and a temporal or 3D based median image. Chapter 3 is self explanatory regarding literature review, the proposed approach for online sunflicker removal, results, and comparison. This sunflicker removal method is used as an optional sub-step in the image preprocessing step of the proposed classification framework.

In Chapter 4, we present the details of our proposed method together with a brief theory of the methods used in different parts. This chapter also explains the implementation details of thematic mapping of the mosaic datasets.

Chapter 5 presents detailed results and a comparative analysis of the proposed method and evaluation algorithms.

The concluding, Chapter 6, summarizes the most important findings and provides guidelines intended for other practitioners to better cope with their own datasets. This chapter also discusses the method's limitations and points out future research directions to address them.

The datasets used in the experiments are explained in Appendix A. Ten different datasets have been used in this thesis including five benthic datasets, three mosaic image datasets, and three texture datasets.

1. INTRODUCTION

Chapter 2

Literature review

In this chapter, a comprehensive summary is presented on the state of the art in computer vision techniques available to do supervised automated annotation of benthic habitats. There are only a few works in the literature that address the issue of underwater object classification with optical imagery. Although some other works have addressed the same problem using acoustic or sonar images, these body of work is not very relevant to our problem of dealing with optical images, given the fundamental difference between optical and acoustic data.

Additionally, we also present in this chapter the standard methods accepted by the community for accessing the classification accuracy of thematic mapping. We describe the theory of the methods used in our method for evaluating the performance of the proposed classification method.

2.1 General framework of object classification

A general supervised approach for object classification in the computer vision community contains standard steps such as image collection, preprocessing, invariant feature extraction (texture, color, shape), feature modification (kernel mapping, dimension reduction, normalization), classifier training and finally testing for accuracy. There are many different computer vision techniques available for each of these steps in the framework. In this thesis, we studied the methods that work best for each step, specially for underwater optical imagery. However, some addi-

2. LITERATURE REVIEW

tional steps can also be adapted to this framework, such as segmentation and post processing. These additional steps are explained in Chapter 4 along with our proposed framework. Figure 2.1 illustrates a general framework for supervised object classification.

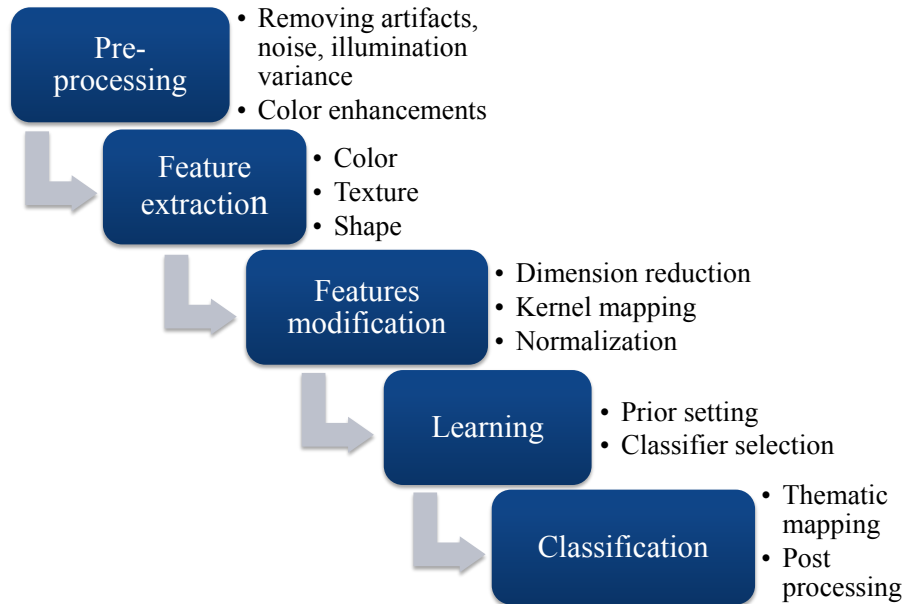


Figure 2.1: General framework for supervised object classification

2.2 Related works on automated benthic habitat classification

Since the last decade, several works have been published that addressed the issue of automated classification of underwater objects using optical imagery. However, none of them could provide consistent performance on various benthic datasets and generally are not accepted as robust.

One of the fundamental references in automated seabed classification using optical imagery is the work by Pican *et al.* [78]. In this work, the grey level co-occurrence matrix (GLCM) [43] or the Kohonen maps [45] are used as texture feature descriptors. Kohonen maps are also used as the classifier in the later part of their implementation. This method presents good accuracy on the 30 classes in their dataset.

However, it is not very robust in terms of scale and rotation variation, illumination and blurring effects of water.

Marcos *et al.* [65; 99] use a feed-forward back-propagation neural network to classify underwater images. They use local binary patterns (LBP) [73] as texture descriptors and normalized chromaticity coordinates (NCC) or mean hue saturation value (HSV) as color descriptor. The limitation of this method is the use of mean HSV or NCC color features which in many cases are not discriminative enough. Also, the neural network needs a large amount of training data to achieve a good learning of the class boundaries, and always need retraining as new samples are added to the training set.

LBP [73] is also used for the identification of crown of thorns starfish (COTS) by Clement *et al.* [9]. This method focuses on just one class, treating the substrate and all other benthic organisms as background class. This method can work only for very distinctly variant classes.

The work by Johnson-Roberson *et al.* [47; 48] employs both acoustic and optical imagery for benthic classification. Acoustic features and visual features are classified separately using a support vector machine (SVM) [12] with assigned weights, which are determined empirically. This work uses a combination of different sensor data, where we depend only on optical imagery [93].

Gleason *et al.* [35] use color and texture features in a two-step algorithm to classify three broad cover types. The main drawback of this system is that it needs a specialized camera capable of acquiring narrow spectral band images, which is specialized and costly hardware. This method is not considered as one of the baseline approaches in our comparison since they do not classify underwater objects based exclusively on standard optical images.

Mehta *et al.* [69] utilize an SVM classifier having pixels as features. Their method were used for coral reef texture classification of only two classes. The method fails on more realistic conditions of noisy images, and under varying illumination conditions. Also, this method does not extract any core features of the class, but rather totally depends on SVM's inherent discriminative power for classification. Without a very large number of training examples generating a good definition of multiple class identity, it would be very difficult to get consistent results with this method.

2. LITERATURE REVIEW

In the work of Pizarro *et al.* [80], 'bag-of-visual-words' [15] features are used. These features are the most commonly used by the computer vision community for object classification. This method attains a good level of accuracy and can be considered as one of the main references in the state of the art methods. For this reason, it is used for comparison against our proposed method in Chapter 5. In their approach [80], an image is represented as a collection of visual words obtained using SIFT (scale invariant feature transform) descriptors [62]. In this method, an entire image is classified as one class or another. Therefore, within-image heterogeneity cannot be classified or quantified. Also, this method requires relatively high resolution images to generate enough interest points that describe the content of the image.

The work by Marcos *et al.* [66] uses LBP [73] and NCC histograms as feature descriptors and a linear discriminant analysis (LDA) [70] as the classifier. The method was tested on only two classes of benthic habitats. Moreover, the LDA classifier is not able to perform consistently well for varied types of seabed image datasets [96]. However, this method is considered as one of the related state of the art methods to be compared against our proposed method.

The work of Stokes and Deane [100] uses normalized color space and discrete cosine transforms (DCT) to classify benthic images. The final classification is done using their proposed probability density weighted mean distance (PDWMD) classifier [100] from the tail of the distribution. This method is time efficient with good accuracy but requires accurate color correction, which may be difficult to achieve on underwater images without controlled lighting. Also, the method uses fixed 64×64 pixel window sizes, which sometimes might not be enough to capture sufficient visual aspects making some classes distinguishable. This work is also considered as one of the state of the art methods that was selected to comparatively analyze the performance of our proposed method.

The work of Padmavathi *et al.* [76] uses kernel principal component analysis (KPCA) [46] to project a SIFT [62] feature vector in a more distinguishable domain. The classification is performed with probabilistic neural network (PNN). Their method achieved high accuracy in their used dataset. However, this method lacks proper comparison with other methods on varied datasets.

The work by Diaz *et al.* [19] uses the local homogeneity coefficient (LHC) [29]

to segment, and pixel by pixel distance of texture features such as energy, entropy and homogeneity to classify. This method can only deal with classes that are highly discriminative by nature, and therefore, is of limited application under water.

Beijbom *et al.* [2] published for the first time, a standard dataset for testing state of the art methods in underwater optical image classification. They also proposed a novel framework for seabed classification which consists of feature vector generation using a maximum response (MR) filter bank [108], and a SVM [12] classifier with a radial basis function kernel. In this method, multiple patch sizes were used, providing a significant improvement relative to classification accuracy. However, the use of multiple patch sizes can be redundant and time consuming. Also, the SVM classifier requires manual tuning to find the parameters that generate the best possible results.

The work by Bewley *et al.* [4] use PCA compression from raw RGB images as local image features and support vector machines as the classifier. This method predicted only the presence or absence of *Ecklonia Radiata* (kelp) on the sea floor. Their method is hardware efficient and works well for kelp. However, their performance is yet to be tested on more varied classes in challenging datasets.

Another approach to underwater object classification can be unsupervised learning, specially for the rare classes where acquiring adequate training samples is very difficult. Bender *et al.* [3] in their recent work, used a novel approach of probabilistic targets least square classifier (PTLSC) to cluster the similar types of areas on the seabed. This method shows promising results and is likely to evolve in future research.

A comparative summary of the existing algorithms for supervised automated underwater object classification using only optical imagery is given in Table 2.1, with entries in bold corresponding to the published methods used for comparison in Chapter 5 of this thesis.

For the cases where the survey images contain enough overlap to allow the extraction of depth information, then 2.5D or even 3D based features can be important as additional features for classification. The work by Friedman *et al.* [31] presented a new method for calculating the rugosity, slope and aspect features of the Delaunay triangulated surface mesh of the seabed terrain by projecting areas onto the plane of best fit using principal component analysis (PCA). They used these features to

2. LITERATURE REVIEW

Table 2.1: A brief summary of methods classifying benthic images. The methods in bold are used in Chapter 5 for performance comparison and are referred to by the underlined authors names. The last column, **N**, contains the number of classes used for testing each method, as reported in their respective papers.

| Authors | Features | Classifiers | N |
|---------------------------------|---|--|-----------|
| Bewley [4] | Texture: Principal component analysis (PCA) from raw RGB image | Support vector machines (SVM) | 2 |
| <u>Beijbom</u> [2] | Texture: Maximum response (MR) filter bank | Support vector machines (SVM) | 9 |
| Padmavathi [76] | Texture: Bag of words using scale-invariant feature transform (SIFT) | Probabilistic neural network (PNN) | 3 |
| <u>Stokes & Deane</u> [100] | Color: (RGB histogram) | Probability density weighted mean distance (PDWMD) | 18 |
| | Texture: discrete cosine transform (DCT) | | |
| <u>Pizarro</u> [80] | Color: normalized chromaticity coordinate (NCC) histogram | Voting of the best matches | 8 |
| | Texture: bag of words using scale invariant feature transform (SIFT) | | |
| | Saliency: Gabor filter response | | |
| Mehta [69] | Color: Pixel intensity | Support vector machines (SVM) | 2 |
| Gleason [35] | Multi-spectral data | Distance measurement | 3 |
| | Texture: grey level co-occurrence matrix (GLCM) | | |
| Johnson-Roberson [47; 48] | Texture: Gabor filter response | Support vector machines (SVM) | 4 |
| | Acoustic: Gabor filter response | | |
| <u>Marcos</u> [65] | Color: normalized chromaticity coordinate (NCC) histogram | 3 layer feed forward back projection neural network | 3 |
| | Texture: local binary pattern (LBP) | | |
| Clement [9] | Texture: local binary pattern (LBP) | Log likelihood measure | 2 |
| Soriano [99] | Color: normalized chromaticity coordinate (NCC) histogram | Log likelihood measure | 5 |
| | Texture: local binary pattern (LBP) | | |
| Pican [78] | Texture: grey level co-occurrence matrix (GLCM) | Kohonen map | 30 |

define the characteristics of the seabed terrain for scientific communities. In our work, we use these 3D statistics [31] as features for classification.

There are several works that use 3D features for classification for general computer vision applications [52; 77]. In the work of Knopp *et al.* [52], Hough transform and 3D Surf were used to generate 3D features. Their method can work well for objects with rigid and structured 3D shape. Paul *et al.* [77] demonstrated that mean curvature and gaussian curvature are the fundamental second-order surface characteristics that possess desirable invariance properties and represent extrinsic and intrinsic surface geometry respectively. Sandbach *et al.* [87] use 3D features for facial expression detection based on predefined shape model of faces. Marton *et al.* [67] proposed the combined use of 2D and 3D features mainly for online robot motion applications. In these mentioned applications, objects that have predefined shape model such as human body, table, etc. are used for classification. In our work we are interested in classifying natural benthic objects that may or may not have very distinctive shape model.

Geographic information systems (GIS) tools like St John BIOMapper use statistics such as curvature, plan curvature, profile curvature, mean depth, variance of depth, surface rugosity, slope, slope of the slope to characterize the complexity of the seafloor. Some of these features can be considered as potential 3D or 2.5D features for underwater object description.

Much additional work has been done on texture classification for other applications besides benthic imagery. Two papers in particular are worth mentioning here, as they reported the highest classification accuracy on standard texture datasets. Hayman *et al.* [44] use filter bank outputs with a SVM classifier that has a kernel based on the chi-square distance. This method works very effectively on low-resolution texture datasets. Zhang *et al.* [110] use an approach that represents images as distributions (signatures or histograms) of features extracted from a sparse set of key points locations. In their work a SVM classifier is trained with kernels based on the earth mover's distance (EMD) and the chi-square distance. We compare the performance of our method on texture datasets with these two methods as well.

The four main methods selected as the benchmark in our comparison are discussed in more detail in the following sections.

2.3 Image based classification

In the image based classification approach, each individual image is classified as belonging to any particular class depending on the visual cues present in the image. The work by Pizarro *et al.* [80] follows this approach. In this approach it is aimed more towards finding the presence of objects in an image rather than finding the location of the objects. For thematic mapping, this method can be used if a small patch around any center pixel is taken and then is classified as any object class.

2.3.1 Pizarro

Pizarro's method *et al.* [80] is an image-based classification method where state of the art 'bag-of-visual-words' [15] object recognition system is used. In this method, an image is converted into a collection of visual words or visual features which can be considered as a discriminative descriptor of that image. A set of training images annotated by experts is used to build a vocabulary of visual words so that the query image can be described by the frequency of occurrence of all these words in the vocabulary. In this system, a class is assigned to the query image, based on the class of the closest training sample. The implemented method by Pizarro is described in the following.

- Initially, the images of both the testing set and the training set are modified by comprehensive image normalization applied independently to each color channel. The normalized color patch is then presented in HSV color space. From only the hue channel, a 24 bin color histogram descriptor is extracted. This histogram contains the color information of the image. Additionally, using SIFT [62], another feature vector is created. Combining these two statistically independent features, a strong descriptor of the image is created.
- Using the final combined feature vector (that have been also normalized), images are presented in a feature dimension that is more discriminative than the raw images. Afterwards, the euclidean distance classification algorithm is used to classify the test images. In their work, 453 images were classified into 8 classes by an expert for training and vocabulary generation.

The method by Pizarro [80] does not perform well for pixel by pixel classified

mapping and can be very inefficient and less accurate if used for making a high resolution pixel by pixel thematic mapping. This method follows the standard image classification approach (defining objects in the image), used by the computer vision community, acquiring moderate performance. However, for underwater images, there are many objects which can only be differentiated using texture information. Texture cues contain discriminative identity of most of the seabed objects. In this method the texture features are defined with the bag of features approach by SIFT [62] descriptors. On flat surfaces, it would be very difficult to acquire enough SIFT points on a featureless image such as sand. Also, the histogram of the Hue channel contains an attenuated color response which is limited in its capability to discriminate between similarly colored object.

2.4 Patch based classification

In patch based classification, a sliding window is used to take patches around the center pixels and classify them based on available visual features. The works by Marcos [65], Stokes & Deane [100] and Beijbom [2] follow this approach.

2.4.1 Marcos

In this method, Local Binary patterns (LBP) are used as texture descriptors and Normalized Chromaticity Coordinates (NCC) or Hue Saturation Value (HSV) as the color descriptor. LBP is used for its robustness to brightness changes and Gaussian blurring and also performs consistently better at recognizing tilted three dimensional textures than other texture descriptors. Using the LBP [73] descriptors and NCC color information, a 14 component feature vector is created.

For the final classification, a supervised three-layered feed-forward, back projection neural network using Matlab neural network toolbox is used where weight updates are done by gradient descent. The Neural Network is trained using the mean square error 0.01 as the learning convergence criterion having 12 hidden units and using momentum term of 0.9.

The method was implemented for only three classes and was compared with the rule based decision tree classification method. The neural network needs a lot of

2. LITERATURE REVIEW

free parameters and can be mathematically complex. It requires more training data to learn a pattern compared with other classification methods. Also, the method has only been tested on three classes with 300 images (640×480 pixels size). This method claims a success rate of 86.5%, which is quite promising.

2.4.2 Stokes & Deane

Stokes & Deane [100] use an approach similar to Marcos *et al.* [65] with different features and classifiers. In their work, normalized color histograms are used as the color descriptor and a discrete cosine transform as the texture descriptor. The classification is performed using a novel approach titled 'probability density weighted mean distance (PDWMD)' from the tail of the distribution [100]. The method is implemented through the following steps.

- The image acquisition is performed having color markers around the frame. Color markers around the frame provide color and intensity consistency between transect images. Color intensity in the frame markers are used to automatically normalize the illumination of all the images as a pre-processing step.
- An annotated image library is created by the experts for training. This library contains 3000 training patches (65×65 pixels in size) belonging to 18 different substrate/organism types. The library of images is converted into feature vectors (DCT and normalized color histograms) for a more compact representation than the raw images.
- For classified mapping, a sliding window of 65×65 pixels (overlapped) is used. For each test patch, the descriptive matrix (or, in other words, feature vectors) is calculated using two dimensional discrete cosines transforms of the image luminance and color histogram (32 bins for each individual color) in RGB color space.
- The novel classification algorithm titled PDWMD considers each class in turn. The distance from the test patch to all the patches in one class in the training set is computed and the mean of minimum three distance is stored. In iteration, this is done for all the classes in the training set. The class with the smallest mean distance is assigned as the label of the test patch.
- Finally, a 3×3 median filter is used as a post-processing step to remove scattered misclassifications.

This method can be considered as one of the most relevant state of the art references for large-scale thematic mapping of the ocean floor. The advantage of this method is that it is very simple to implement and also time efficient with good accuracy. The limitation of the method is it characterizes texture using only the DCT of Luminance of the image patch, whereas other available texture features, such as GLCM and LBP among others, can be more robust and accurate. The weighting of the color features and texture features has been selected empirically. This method works well if the testing and the training images are taken in almost the same conditions in terms of water depth, distance between the object and the camera, and water quality.

2.4.3 Beijbom

In the most recent work of Beijbom *et al.* [2], they proposed a novel framework for classification which comprises an MR filter bank [85] using Texton mapping and a LIBSVM classifier with a radial basis function kernel. In this method, multiple patch sizes are used providing a significant improvement in terms of performance. Preprocessing was done using color channel stretching in the Lab color space [2]. In order to obtain the Texton map for the specific dataset, a subset of the images is exclusively used to evaluate the filter responses from each of the classes. K-means clustering is applied to all these filter responses to create the texture dictionary.

This method also follows the patch by patch classification approach. The texture filter bank contains very reliable descriptors of the texture contents of the objects to be classified. They reported a good classification accuracy with their published datasets. However, their method deals with scale variation by using multiple patch sizes, which can be redundant. Also, they used a standard SVM classifier that needs tuning of the parameters to get the best result out of any specific dataset.

2.5 Accuracy assessment methods

There are several methods popularly used by researchers for the quantitative assessment of thematic mapping (also referred to as spatial simulation model [13; 28]). In general, the confusion matrix (referred to as error matrix [61] or contingency

2. LITERATURE REVIEW

matrix [13]) contains all the information to calculate quantities for quality assessment. Among these quantities, overall accuracy [101], kappa [10], tau [71] and the average mutual information (AMI) [26] can be considered as the most common and accepted ones. Moreover, precision and recall curves, receiver operating characteristics (ROC), average precision and the area under the ROC curve are popularly used in the computer vision literature for the assessment of the classification quality. All of the above indicators present a specific point of view on the quality measure related to thematic mapping.

In the following section, the theoretical background of the confusion matrix [61], overall accuracy (OA) [101], kappa (K) [10], tau (T) [71], average mutual information (AMI) [26] are discussed in brief.

Confusion matrix

The confusion matrix is a special matrix containing, as rows, the particular classes by one classification scheme and, as columns, the results of other classification scheme. One of the classification results can be considered as a reference (or ground truth, in many cases) and is normally placed in the columns. This matrix is most popularly used for its effective representation capability of classification accuracy in global as well as an individual class sense.

Let us assume that n samples were distributed through k^2 cells in which each sample is associated to one of the k classes within the simulation classification (usually rows), and independently, to one of the k classes within the reference data (usually columns). Thus, n_{ij} is the number of samples classified in class i ($i = 1, 2, \dots, K$) in classification and class j ($j = 1, 2, \dots, k$) within the set of reference data (Table 2.2). Here n_{i+} is the number of samples classified within class i of the simulation classification, and n_{+j} is the number of samples within class j in the reference data.

$$n_{i+} = \sum_{j=1}^k n_{ij} \quad n_{+j} = \sum_{i=1}^k n_{ij} \quad (2.1)$$

P_{ij} is described as the proportion of the samples within cell i, j corresponding to n_{ij} , i.e. $P_{ij} = n_{ij}/n$. P_{i+} and p_{+j} are defined by

Table 2.2: A numerical definition of a confusion matrix

| | | Input | | | | |
|--------|-----------|----------|----------|----------|-----|-----------|
| | | class 1 | class 2 | class 3 | ... | class k |
| Output | class 1 | n_{11} | n_{12} | n_{13} | ... | n_{1k} |
| | class 2 | n_{21} | n_{22} | n_{23} | ... | n_{2k} |
| | class 3 | n_{31} | n_{32} | n_{33} | ... | n_{3k} |
| | ... | ... | ... | ... | ... | ... |
| | class k | n_{k1} | n_{k2} | n_{k3} | ... | n_{kk} |

$$p_{i+} = \sum_{j=1}^k p_{ij} \quad p_{+j} = \sum_{i=1}^k p_{ij} \quad (2.2)$$

Overall accuracy (OA)

Overall accuracy (OA) is the sum of the major diagonal in the confusion matrix (i.e. the correctly classified units) divided by the number of units in the whole matrix. n is the total number of samples.

$$OA = \frac{\sum_{i=1}^k n_{ii}}{n} \quad (2.3)$$

Kappa

The kappa (K) statistics for quality assessment is a commonly used indicator in remote sensing fields to compare thematic map accuracies. The kappa value is the measurement of how the classification result relates to the reference data. This measurement discards the chance agreement from the overall accuracy [10]. P_o is the proportion of cases in agreement (correctly classified) and P_c is the proportion of agreement that is expected by chance. The following equations are used to calculate the kappa coefficient.

2. LITERATURE REVIEW

$$p_o = \sum_{i=1}^K p_{ii} \quad p_c = \sum_{i=1}^K p_{+i}p_{i+} \quad (2.4)$$

$$K = \frac{p_o - p_c}{1 - p_c} \quad (2.5)$$

To analyze the agreement for an individual class inside the matrix, the conditional Kappa coefficient (K_c) can be used.

$$K_c = \frac{nn_{ii} - n_{i+n+i}}{nn_{i+} - n_{i+}n_{+i}} \quad (2.6)$$

To assign different weights to different types of errors, the weighted kappa, K_w can be used. This allows more penalization of some specific errors that are deemed most important. w_{ij} is the weight associated to cell ij in the matrix. The proportion p_{ij} of the ij cell is affected by the weight, w_{ij} . The weights should be limited to $0 \leq w_{ij} \leq 1$ interval for $i \neq j$.

$$P_o^* = \sum_{i=1}^k \sum_{j=1}^k w_{ij} p_{ij} \quad P_c^* = \sum_{i=1}^k \sum_{j=1}^k w_{ij} p_{i+} p_{+j} \quad (2.7)$$

$$K_w = \frac{p_o^* - p_c^*}{1 - p_c^*} \quad (2.8)$$

Tau (T)

The Tau coefficient (T) measures the classification accuracy in relation to the random adjustment of pixels to classes [71]. This is similar to the kappa coefficient (K) but uses the prior probability of individual classes. This information is not present in the confusion matrix.

$$p_i = \frac{x_i}{n} \quad p_r = \sum_{i=1}^k p_i + p_i \quad (2.9)$$

$$T = \frac{p_o - p_r}{1 - p_r} \quad (2.10)$$

Average mutual information (AMI)

Using the information theory, we can measure the information gain after classification by comparing it with the reference data. With the overlap of map A (reference data) and map B one can draw up a contingency table $c_1 \times c_2$ with c_1 rows pertaining to map A and c_2 rows pertaining to map B . Each element of the matrix (n_{ij}) corresponds to the number of pixels which belong to class j on map A and to class i on map B . The total number of elements of the matrix is represented by n . The probability of an element belonging to class j on map A and class i on map B , $p(a_j, b_i)$, is obtained by dividing the element of the contingency table, n_{ij} by the total number of elements (n) in the table.

The amount of information that map A contains on map B is reflected in the reduction of uncertainty relative to map A when map B is known. This can be expressed as conditional probability, $p(a_j/b_i)$, which is the probability of a pixel belonging to class j on map A when the pixel is of class i on map B . From probability theory,

$$p(a_j/b_i) = \frac{p(a_j, b_i)}{p(b_i)} \quad (2.11)$$

$$AMI = \sum_{i=1}^k \sum_{j=1}^k p(b_i, a_j) \log \left[\frac{p(b_i, a_j)}{p(b_i)p(a_j)} \right] \quad (2.12)$$

The measurement based on the information theory of the sharing of information,

2. LITERATURE REVIEW

AMI, is applied to the comparison of thematic maps. AMI measures consistency instead of correction [13; 28]. Intuitively, AMI measures the information that maps A and B share. For example, if map A and map B are independent, then map A does not give any information about map B and vice versa, therefore AMI is zero. However, if map A and map B are similar, then all the information conveyed by map A is shared with map B. In this case, the AMI is the same as the uncertainty of map A or map B alone.

2.6 Summary

In this chapter, the most relevant related work on underwater image classification has been presented. The works of Pizarro [80], Marcos [65], Stokes & Deane [100] and Beijbom [2] are the main ones used for comparing the performance of our proposed method. However, the existing methods have only been tested on unique image datasets in the papers in which they were published. Without comparison on standard datasets, it is impossible to assess the relative effectiveness and efficiency of these methods.

Chapter 3

Online Sunflicker Removal Using Dynamic Texture Prediction

The sunflicker effect can be modeled as a dynamic texture due to its properties. The Open Loop Linear Dynamic System (OLLDS) (dynamic texture modeling algorithm) by Doretto *et al.* [20] is used in this method to learn the sunflicker pattern in a dataset. OLLDS is a novel way of defining dynamic texture as a second-order stationary process with an arbitrary and precise covariance sequence. Once the dynamic texture model is learned using the first few images of a sequence, an approximation of the next dynamic texture image of that sequence can be predicted. Using this approach, prediction of the sunflicker effect (as a dynamic texture) allows for coarsely removing it from the next image in the sequence and, thereby registering the current image against the next image with better accuracy. We found comparatively better performance of this approach with respect to the closest related method in the literature by Gracias *et al.* [37]. Also, removing sunflicker from the raw images helps the classification method to achieve better accuracy; due mainly to the higher information content on the object class in sunflicker free images.

3.1 Background

Among the many other unique challenges of underwater optical systems, sunflicker is one of the most prominent effects in shallow water on sunny days. The sunflicker

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

effect comes from refracted sunlight casting fast moving patterns on the seafloor. Figure 3.1 illustrates the way in which sunflicker corrupts an image's appearance, altering human perception of the scene and affecting the natural performance of image processing algorithms. Figure 3.2 shows an example of fast changes in the appearance of underwater objects in consecutive images acquired at a short time interval. In this chapter, our proposed novel method for online sunflicker effect removal is presented. This method can be considered an optional sub-step in the image preprocessing step of the proposed classification framework. This step can help extract more reliable information about the object's characteristics, which can aid the classification performance with a higher accuracy gain. Figure 3.3 illustrates a comparative example of a mosaic creation before and after sunflicker removal.

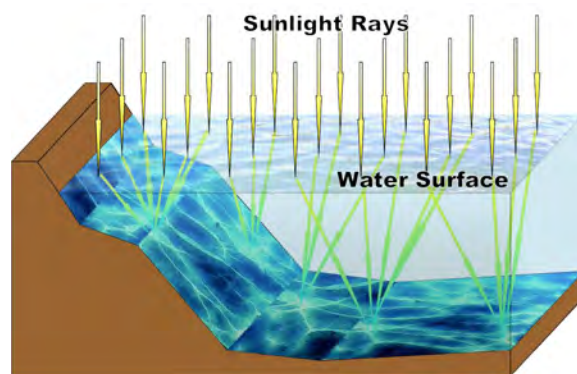
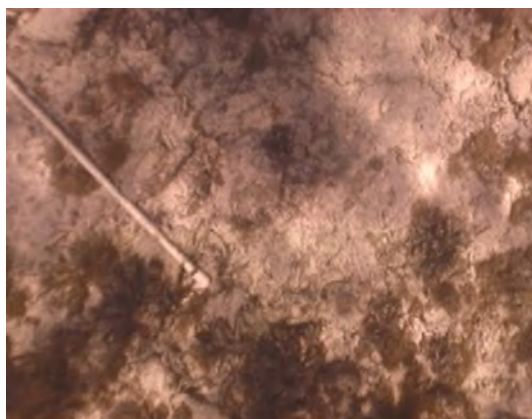


Figure 3.1: Refracted sunlight creating illumination patterns on the sea floor, varying in space & time following the dynamics of surface waves.



(a) Sequence 1



(b) Sequence 2

Figure 3.2: Example of rapid changes in the appearance of underwater objects in consecutive images acquired in a short time interval.

3. Online Sunflicker Removal Using Dynamic Texture Prediction

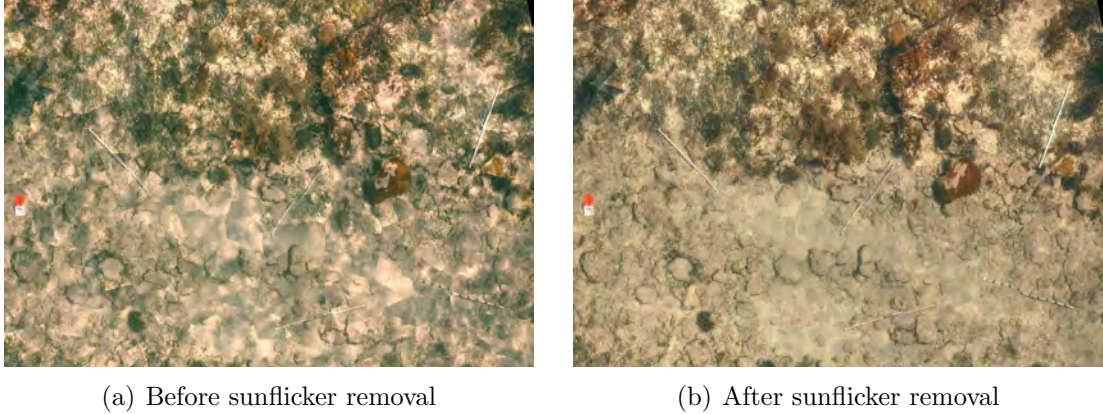


Figure 3.3: Effects of sunflicker removal in a mosaic creation.

3.2 Related work

Combining the work of Weiss [109] on removing shadows in land scenes and the work of Matsushita *et al.* [68] on shadow elimination in surveillance video, Schechner and Karpel [89] proposed an approach for removing sunflicker in underwater images. This approach uses the observation that the spatial intensity gradients of the caustics tend to be sparse. They calculate a gradient field (robust to illumination changes) from a temporal median over the gradients of a small number of images. By integrating this median gradient field, the sunflicker is finally removed. However, camera motion is not considered in this method.

Using the information exchange theory, two transparent overlapped videos can be separated as shown in the work by Sarel and Irani [88] and in the work by Ukrainitz and Irani [105]. This approach works under the assumption that one of these transparent videos contains a repetitive dynamic sequence. The method can handle repetitive dynamic sequence variance to a certain degree which is not precisely quantified. In this approach, a large number of frames from the same camera position are required to grab a complete cycle of the repetitive dynamic sequence, making it almost impossible for moving cameras.

Another interesting approach to deal with the same challenge is to use polarization information [90]. In this approach, the assumption is that the refracted sunlight has unique polarization characteristics. With the help of these variant polarization characteristics it becomes possible to separate the sunflicker layer from the original

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

image. However, to acquire polarization information, special cameras with polarized filters or imaging sensors are required, which increases the cost and complexity of the image acquisition system. Such special systems are not commonly deployed. Candes *et al.* [21] recovers the low rank matrix from the image sequence which represents the illumination field. However, this method only works when there is no camera motion.

The work by Gracias *et al.* [37] addresses the removal of sunflicker considering camera motion. The method works based on the assumption that video sequences allow several observations of the same area of the seafloor over time. Under such conditions, computing the differences in images between a given reference frame and the temporal median of a registered set of neighboring images is possible. This difference image has two components with separable spectral content. One is related to the illumination field (which has lower spatial frequencies) and the other to the registration inaccuracies (mainly having higher frequencies). By filtering out the high frequency component of the difference image, the illumination field can be approximately recovered. The main limitation of this approach is that the median image for each frame is obtained from both past and future frames in a non-causal way, i.e. future frames are required to process the current image. Moreover, the computation of the temporal median requires the original images to be correctly registered before removing the sunflicker, which may be impossible in cases where strong sunflicker is present.

3.3 Framework

Our proposed method is an extension of the work by Gracias *et al.* [37] adding 3D information in creating the median image and converting the method into an online (causal) system [94]. We use dynamic texture modeling and synthesizing [20] to approximate the sunflicker pattern of the current frame from a few previous frames. The presented approach attains a higher registration performance even under heavy illumination fluctuation. Also, this method is strictly causal (*i.e.* does not rely on future observations), fulfilling the important condition of online operation required for visual-based robot navigation.

3. Online Sunflicker Removal Using Dynamic Texture Prediction

3.3.1 Dynamic texture modeling and synthesizing

The OLLDS model can be used for extrapolating synthetic sequences of any duration at negligible computational cost. The underlying assumption in this approach is that the individual images are the realizations of the output of a dynamic system driven by an independent and identically distributed (IID) process. A stochastic process is stationary (of order k) if the joint statistics (up to order k) are time-invariant. For instance a process $I_m(t)$ is second-order stationary if its mean $I_m = E[I(t)]$ is constant and its covariance $E[(I(t_1) - I_m)(I(t_2) - I_m)]$ only depends upon $t_2 - t_1$. Under the hypothesis of second-order stationarity, a closed-form sub-optimal solution of the learning problem obtained as follows:

1. A linear dynamic texture is modeled as an auto-regressive moving average process (ARMA) with unknown input distribution, in the form,

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + z(t) \quad (3.1)$$

$$\mathbf{y}(t) = \mathbf{O}\mathbf{x}(t) + w(t) \quad (3.2)$$

where $\mathbf{y}(t)$ is the observation vector, in this case the image sequence; $\mathbf{x}(t)$ corresponds to the hidden state vector; \mathbf{A} is the system matrix; \mathbf{O} is the output matrix and $z(t)$, $w(t)$ are Gaussian white noises.

2. Taking the SVD of $\mathbf{y}(t)$, $\mathbf{x}(t)$ and \mathbf{O} can be found from the following equations. Here τ is the number of images used to learn the dynamic model of the system.

$$\mathbf{y}_1^\tau = U\Sigma V^T \quad (3.3)$$

$$\mathbf{O}(\tau) = U \quad (3.4)$$

$$\mathbf{x}(\tau) = \Sigma V^T \quad (3.5)$$

where \mathbf{y}_1^τ is the observation vector from image instance 1 to τ of the image sequence used to create the model; $\mathbf{O}(\tau)$ is the output matrix at instance τ ; $\mathbf{x}(\tau)$ corresponds to the hidden state vector at instance τ .

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

3. Matrix \mathbf{A} can be determined uniquely by

$$A(\tau) = \Sigma V^T D_1 V (V^T D_2 V)^{-1} \Sigma^{-1} \quad (3.6)$$

where

$$D_1 = \begin{bmatrix} 0 & 0 \\ I_{\tau-1} & 0 \end{bmatrix} \quad D_2 = \begin{bmatrix} I_{\tau-1} & 0 \\ 0 & 0 \end{bmatrix}$$

3.3.2 Motion compensated filtering

Let us consider a set of registered images. We refer to a given image by the discrete parameter t which indexes the images temporally. The radiance L of a given pixel with coordinates (x, y) can be modeled as

$$L_t(x, y) = E_t(x, y) \cdot R_t(x, y) \quad (3.7)$$

where E_t is the irradiance of the sunlight over the 3D scene at the location defined by pixel (x, y) at time t , after absorption in the water, and $R(x, y)$ is the bidirectional reflectance distribution function. For underwater natural scenes, where diffuse reflectance models are applicable, R is assumed to be independent of both light and view directions.

Converting the expression for L_t to a logarithmic scale allows the use of linear filtering over the illumination and reflectance.

$$l_t(x, y) = e_t(x, y) + r_t(x, y) \quad (3.8)$$

For approximately constant water depth and realistic finite cases, the median converges significantly faster to an average value of l_t than the sample mean. We name this median image as temporal median through rest of the thesis. Let us consider the median of the radiance over an interval $[t_0, t_1]$.

$$I_{med}(x, y) = med_{[t_0, t_1]} I_t(x, y) \approx e + r_{med}(x, y) \quad (3.9)$$

Here $r_{med}(x, y)$ stands for an approximation of the median of reflectance. The difference $d_t(x, y)$ of a given image $l_t(x, y)$ with the median radiance $l_{med}(x, y)$ is

3. Online Sunflicker Removal Using Dynamic Texture Prediction

used to recover the approximate background image.

$$d_{l_t}(x, y) = l_t(x, y) - I_{med}(x, y) \quad (3.10)$$

This difference $d_{l_t}(x,y)$ has two main components. The first component relates to the instant fluctuation of the illumination field and has lower spatial frequencies. The second component relates to inaccuracies in the image registration and has higher spatial frequencies. After applying a low pass filter to the difference image, the low frequency components that resemble the illumination field are only kept in the output. This approximated illumination field is later used to correct the main input image, recovering a flicker free version of the image.

3.3.3 Median image using 3D registration

Although the motion compensating filtering approach for sunflicker removal is robust to a relatively high level of registration inaccuracies, it starts failing when the spacial frequencies of the sunflicker patterns are no longer separable from the blurriness induced by poor registration (when computing the median). Moreover, the registration quality can be seriously degraded for terrains with high 3D content, especially if a planar motion model is used for the image registration [37]. For this reason we have developed and implemented a 3D estimation and registration procedure using a computationally light-weight method. In our work, the 3D structure is sparsely estimated, inspired by the method of Rother and Carlsson [85]. However, our approach for 3D structure estimation differs from this method as it does not require the explicit presence of a plane. Our method starts with an initial registration of images using 2D homographies from where the camera rotations are estimated. With the camera rotations knowledge, the problem of finding the locations of the camera centers and the locations of the 3D points becomes linear. This is solved with the singular value decomposition (SVD) to obtain a least square (LS) solution. For interest point detection and outlier rejection, Scale Invariant Feature Transform (SIFT) [62] and RANdom SAmple Consensus (RANSAC) [27] are used, respectively. The linear 3D object model created from several neighboring images is projected onto the reference images. From these projected images and the reference

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

image, a median image is calculated and considered the 3D-based median image. Figure 3.4 shows examples of six projected neighboring images on the reference image. Figure 3.5 shows an example of a 3D-based median image created with the help of images from Figure 3.4.

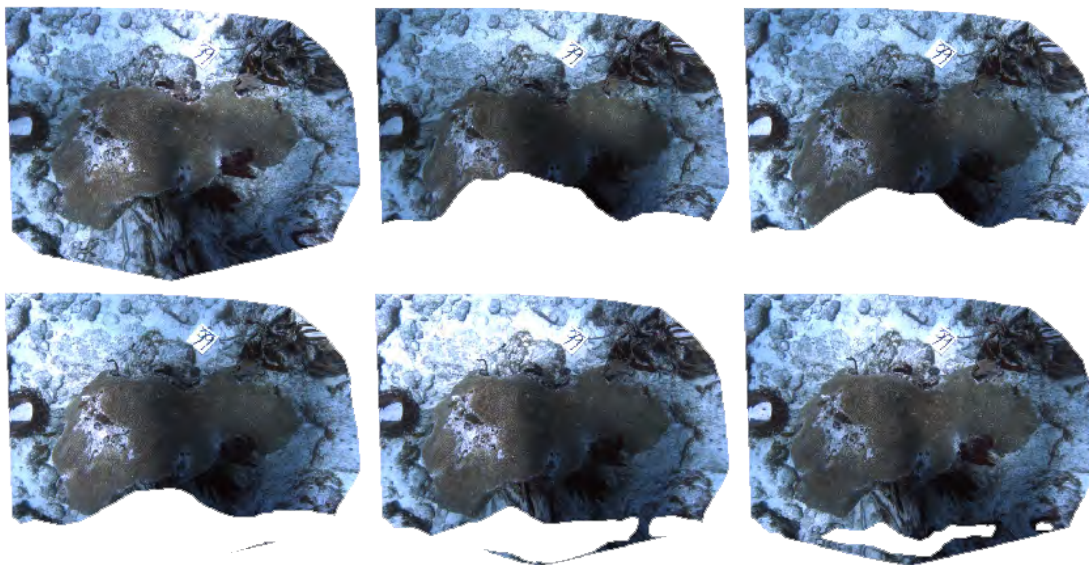


Figure 3.4: Projected texture mapping on the *reference* images from different neighboring images. We can observe in each image that only the common areas on the seabed seen from both camera positions (neighboring image and reference image) are only projected on the reference image plane.

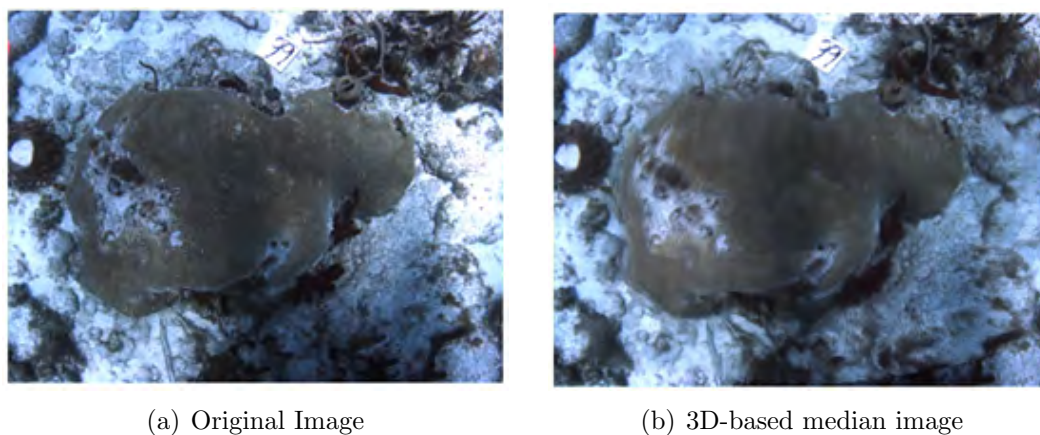


Figure 3.5: Example of created 3D-based median image. We can observe that the 3D-based median image contains motion blur due to camera movement.

3.4 Proposed algorithm

In the online sunflicker removal method proposed in this work, the approximate illumination field of the current frame is found in the dynamic texture prediction step. The homography is mainly calculated between the temporary recovered current image (using the approximate illumination of the current frame) and the last recovered image. Finally, the new image can be recovered by using a median image. The median image is created using the 3D based method described above. If it fails, the temporal median is used instead. The steps are illustrated in Figure 3.6.

The proposed approach considers the following assumptions to be valid:

- Illumination field is a dynamic texture
- Camera movement in the video sequence is smooth
- Bottom of the sea is approximately flat

The main algorithm contains several steps which are presented next, using the following nomenclature:

- $I_{0,k}$ - Original input image obtained at time instant k
- $I_{R,k}$ - Recovered image obtained from $I_{0,k}$ after sunflicker removal at time instant k
- $I_{D,k}$ - low pass filtered version of the difference image at time k (used as estimate of the illumination field)
- $I_{M,k}$ - Median image obtained from N frames after being warped onto the reference image frame $I_{0,k}$
- N - the number of images present in the learning sequence
- $H_{k,k-1}$ - Homography relating the image frames at times k and $k - 1$

The algorithm comprises the following steps:

1. Apply the motion compensated filtering method [37] to register and recover the first few images from the sun flickering effects. In our system, we use the first 25 frames for this step.

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

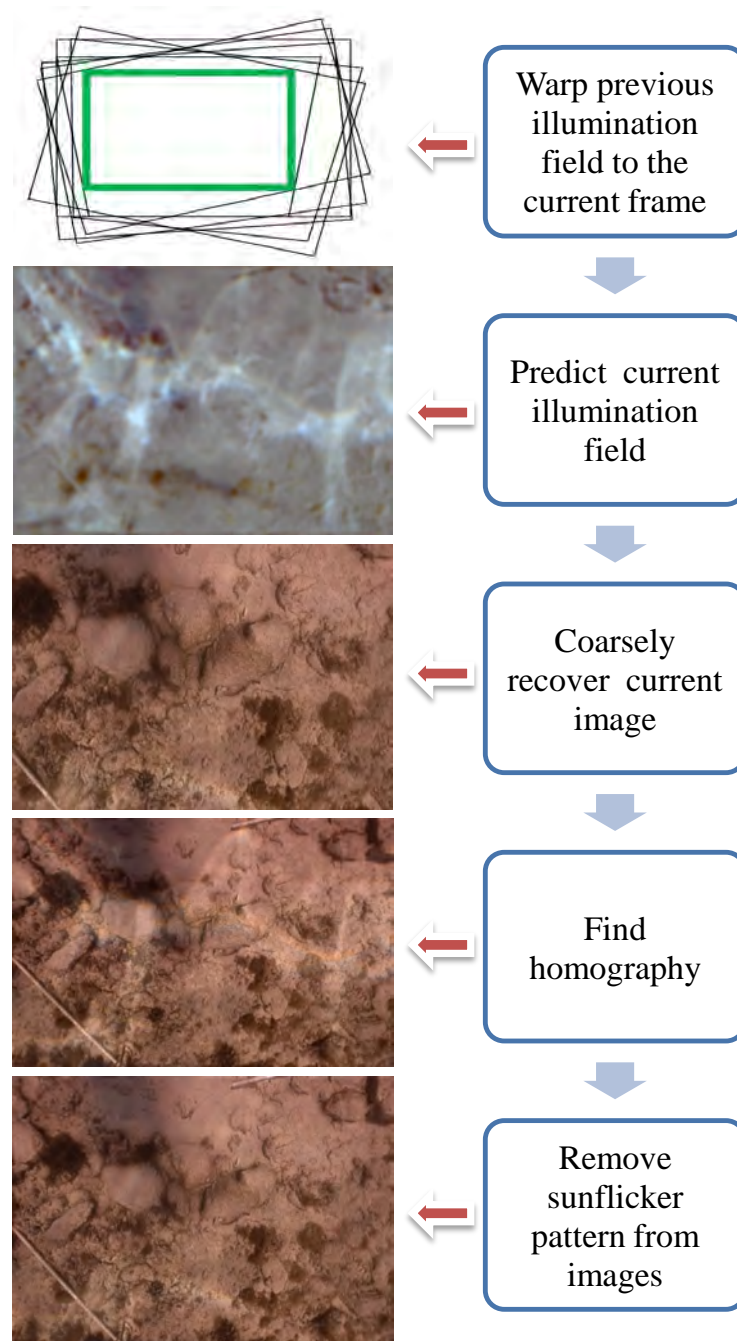


Figure 3.6: Step-by-step flow diagram of the proposed online method for sunflicker removal. From top to bottom, (1) warping previous illumination field to the current frame, (2) predicting the current illumination field, (3) coarsely recovering the current image, (4) finding homography between the current and the previous frame and (5) removing the sunflicker pattern from the image using the calculated homography

3. Online Sunflicker Removal Using Dynamic Texture Prediction

2. Get the new image in the sequence $I_{0,k}$ assuming that the previous images $I_{0,k-1}, \dots, I_{0,k-N}$ have been recovered after sunflicker removal ($I_{R,k-1}, \dots, I_{R,k-N}$). Advance to time instance k (i.e. all previous data structures that had index k now have index $k - 1$).

3. Predict the flicker pattern by:

- Warping all the filtered version of the difference images, $I_{D,k-1}, \dots, I_{D,k-N}$ with respect to the current frame $I_{0,k}$ (to be recovered). Assume that $H_{k,k-1} \approx H_{k-1,k-2}$. All other previous homographies were obtained from actual image matches and thus previously known.
- Learning the sunflicker pattern from the registered filtered difference images. In the learning phase, all the difference images in the previous frames I_D ($I_{D,k-1}, \dots, I_{D,k-N}$) are converted into a column matrix. Using the array of all the registered difference image, a large matrix \mathbf{W}_{t-1} is created having P rows and N columns. P is the number of pixels per frame and N is the total number of frames in the learning sequence.

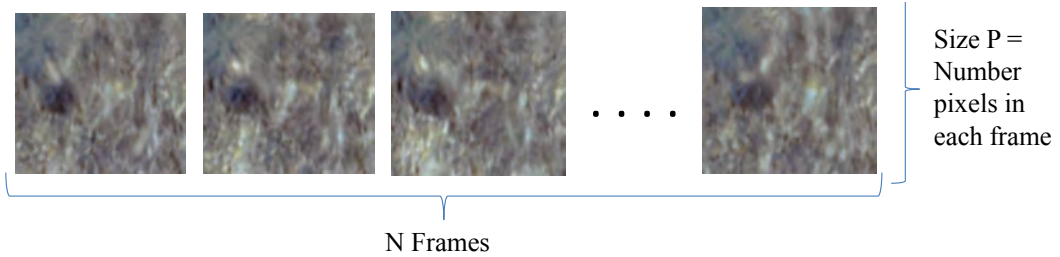


Figure 3.7: Illustration of \mathbf{W}_{t-1} matrix construction using the difference images of the *Grounding* dataset.

- Predicting the $I_{D,k}$ using the learned model. For learning, an open loop linear dynamic model [20] is used. In this step the last frame $I_{D,k-1}$ of the learned sequence is considered as the first frame for synthesizing the approximate next frame (here termed as prediction).
4. Create the approximate sunflicker recovered image using the predicted low pass filtered version of the difference image $I_{D,k}$. This recovered image is denoted by $\hat{I}_{R,k}$ and this predicted illumination field is denoted as $\hat{I}_{D,k}$.

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

$$\hat{I}_{R,k}(x, y) = l_{0,k}(x, y) - \hat{I}_{D,k}(x, y) \quad (3.11)$$

5. Perform image registration between $\hat{I}_{R,k}$ and $I_{R,k-1}$. From this, obtain the current homography $H_{k,k-1}$.

6. Calculate $I_{M,k}$. A median image is created by the 3D based method described earlier. If it fails, the temporal median is used instead.

7. Using $I_{M,k}$, find the high frequency filtered difference image for the current frame, $I_{D,k}$. At the end, obtain the final recovered image $I_{R,k}$ using the following equation.

$$I_{R,k}(x, y) = l_{0,k}(x, y) - I_{D,k}(x, y) \quad (3.12)$$

8. Go to step 2 and do the same for the next frames.

Figure 3.8 shows an example of how, in each step, the image is coarsely recovered, registered, and finally, the sunflicker is removed. The top left shows the original input image, $l_{0,k}$ and the bottom left shows the real illumination field, ($I_{D,k}$). In the bottom center, the predicted illumination field, $\hat{I}_{D,k}$ is shown. The upper center represents the intermediate condition of the recovered image, $\hat{I}_{R,k}$. The top right is the final recovered image, $I_{R,k}$ and the bottom right is the final median image, $I_{M,k}$.

These steps are applied over each color channel independently. Strong caustics lead to overexposure and intensity clipping in one or more of the color channels, resulting in chromaticity changes in the original images. These clippings typically affect different regions of the images over time, given the non-stationary nature of the caustics. The median is not affected by these transient clippings, whereas the average is. The low pass filtering is performed using a fourth order Butterworth filter [55], with a manually adjusted cutoff frequency.

3. Online Sunflicker Removal Using Dynamic Texture Prediction

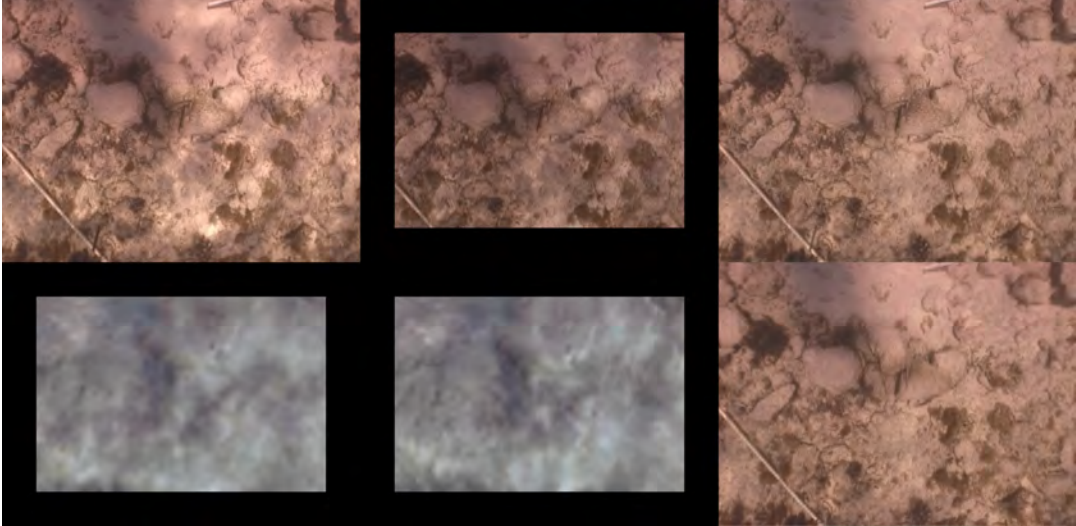


Figure 3.8: Step by step results of online sunflicker removal for the *Grounding* sequence. The top left corresponds to the original input image, $l_{0,k}$ and bottom left is the original illumination field in the input image, $I_{D,k}$. In the top middle, the temporary recovered image, $\hat{I}_{R,k}$ is shown and the bottom middle corresponds to the predicted illumination field, $\hat{I}_{D,k}$. In the top right, the final recovered image, $I_{R,k}$ is illustrated and in the bottom right we show the final median image, $I_{M,k}$

Due to camera motion, the stack of warped difference images described in step 3, may not cover the entire area of the current frame. If one considers the whole area of the current frame, this creates a condition of missing data in the \mathbf{W} matrix for PCA. To circumvent this condition, the part of the area present and valid in each warped frame at current frame location is considered.

3.5 Sunflicker datasets

This section describes the video sequence datasets used to evaluate the proposed method. All the video sequence in the datasets have 150 frames at a 12 frames per second speed. In the results, performances over the last 125 frames are shown. Only the first 25 frames are used for initialization as described in step one of the proposed method. Figure 3.9 shows examples of single frames from the four natural video sequence datasets used in this work.

Grounding sequence

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION



(a) *Grounding* sequence



(b) *Rocky* sequence



(c) *Andros* sequence



(d) *Uneven* sequence

Figure 3.9: Examples of single frames from the *Grounding*, *Rocky*, *Andros* and *uneven* sequences

3. Online Sunflicker Removal Using Dynamic Texture Prediction

In 2002, the 49-foot vessel Evening Star ran aground on a hard bottom full of stones and soft corals in the Biscayne National Park, in Florida (2523.332 N, 8009.874 W, 3 m depth). The *Grounding* sequence (Figure 3.9(a)) video captured the scar created during this incident using a Sony TRV900 DV camcorder placed in an underwater housing. A digital depth gauge was used by the camera operator to keep a consistent depth during the surveys. This sequence was affected by sunflicker in most of the frames. We used 150 frames of 720 by 530 pixels.

Rocky sequence

The *Rocky* sequence (Figure 3.9(b)) which is more challenging than the *Grounding* sequence, was captured in shallow waters with a very rocky bottom. This sequence was also severely affected by sunflicker in all of the frames. The frame size in this sequence is 720 by 530 pixels.

Andros sequence

This is the most challenging sequence of the five datasets, acquired in very shallow waters of less than 2 meters under intense sunlight. In this video, the illumination patterns have simultaneously very high spatial and temporal frequencies. An example is given in Figure 3.9(c). Each frame in this sequence is 720 by 530 pixels in size.

Uneven sequence

This particular video sequence has good depth variation suitable for 3D reconstruction. It contains frames of 1024 by 768 pixels in size. An example is given in Figure 3.9(d).

Synthetic sequence

Aiming for a quantitative performance evaluation of the proposed online sunflicker removal method, we created a synthetic sunflicker affected video sequence. The sunflicker pattern captured in a swimming pool video is projected onto the sunflicker removed grounding video sequence to create this *Synthetic* sequence. Both

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION

sequences, combined here, present approximately the same distance between the camera and the scene. An example of sunflicker pattern extracted from the pool video is given in Figure 3.10. The size of the frames in this sequence is 1360 by 726 pixels.

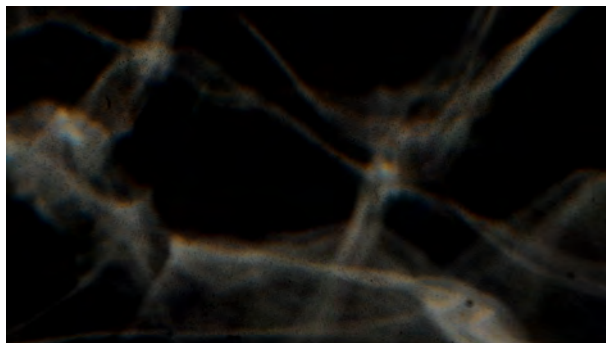


Figure 3.10: Sunflicker pattern extracted from a swimming pool video.

3.6 Methodology

The performance evaluation and comparison of the proposed approach with the closest method in the literature [37] was done on the test datasets presented above, which have distinct refracted sunlight conditions (*Grounding*, *Rocky*, *Andros*, *Uneven* and *Synthetic* sequences). The main evaluating criteria is the number of inliers found per time-consecutive image pair in each registration step. This criterion was found to be a good indicator of the image de-flickering performance. The better the sunflicker removal achieved, the larger the number of inliers per time-consecutive image pair, assuming that all the other influencing factors are constant.

We used a maximum grid intensity variation measurement to assess the performance of the median image creation. If the 3D-based median image creation method fails due to lack of correspondences, the system switches to the temporal median image approach. For maximum grid intensity variation calculation, the reference image and the created median image are divided into a grid of 6 by 6 cells and the average pixel intensity difference is calculated for each cell in the grid. The maximum absolute average difference is finally used for the median image quality (the image quality is inversely proportional with the distance).

3.7 Results

The proposed method outperforms the previous method in almost all the frames on the *Grounding* sequence as shown in Figure 3.11. Quantitatively, the matching performance is improved by 46% based on the inliers detected in every pair of frames. Figure 3.11 provides a comparative analysis of the proposed online method and the offline method of [37] in terms of the number of inliers detected during the registration step. Some examples of the recovered frames for different video sequences are given in Fig. 3.12.

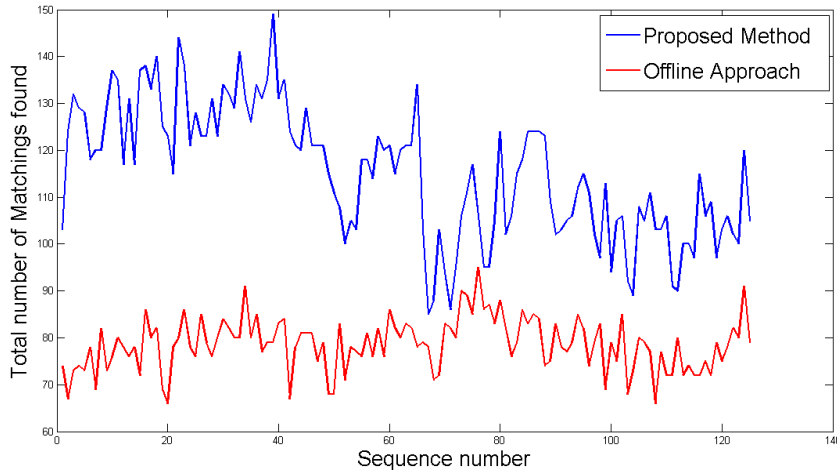


Figure 3.11: Comparison between the proposed and the offline method [37] for the *Grounding* sequence

A comprehensive comparison of the matching performance between the online and offline sunflicker removal methods using both homography and 3D-based median images with the *Grounding* sequence is shown in Figure 3.13. In this particular sequence, the use of 3D-based median image showed an unsatisfactory performance for both the online and the offline sunflicker removal methods (green and blue line). In the case of temporal median image, both the online and offline method achieved better accuracy and stability.

Figure 3.14 shows the comparative analysis of the *Rocky*, *Andora* and *Synthetic* sequences. For the *Rocky* sequence, the proposed online method performs significantly better than the offline method in terms of number of inliers per registration (Figure 3.14(a)). The overall gain in the matching performance for the proposed method is about 67% compared to Gracias *et al.* [37]. For the *Andros* sequence, (see

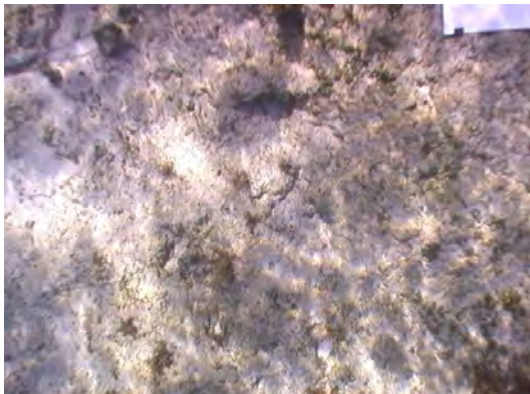
3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION



(a) Original Image



(b) Recovered Image



(c) Original Image



(d) Recovered Image



(e) Original Image



(f) Recovered Image

Figure 3.12: Illustration of the online sunflicker removal performance

3. Online Sunflicker Removal Using Dynamic Texture Prediction

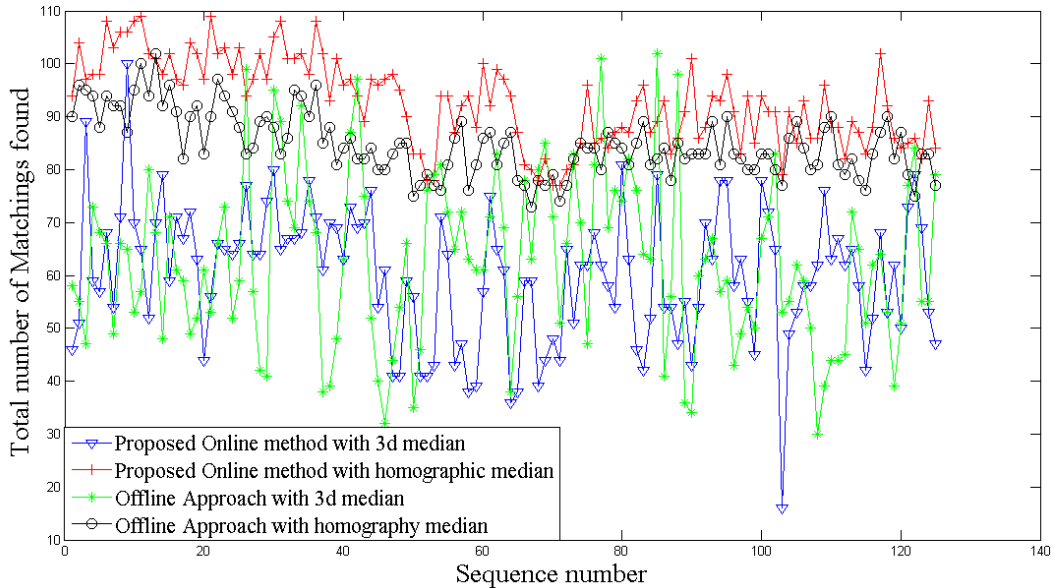


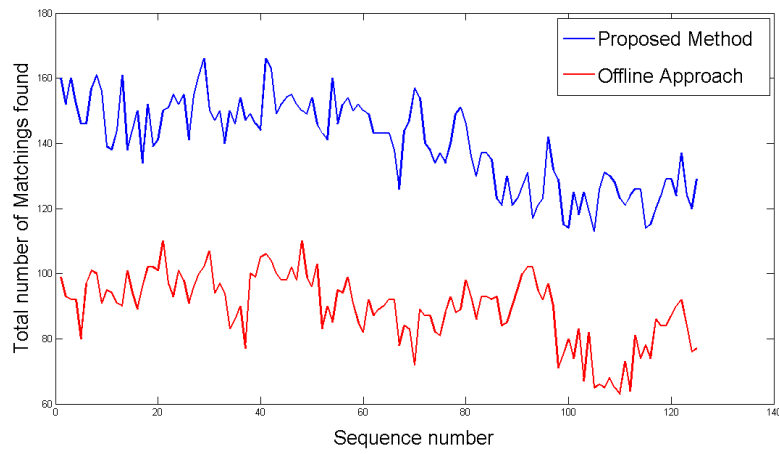
Figure 3.13: Performance comparison with the *Grounding* video sequence

Figure 3.14(b)), the proposed method performs 15% better. We also compare the online and the offline sunflicker removal methods on the *Synthetic* video sequence to numerically quantify the quality of the de-flickering. As in Figure 3.15, the online method has less error and recreates more the flicker free ground truth more accurately.

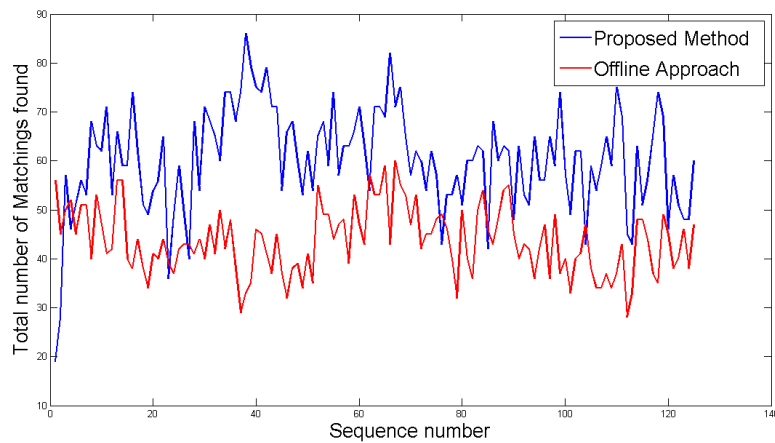
Figure 3.16 is an example of the offline sunflicker removal performance on the *Uneven* video sequence using both 3D-based median and homography median images. The 3D-based median image fails to perform consistently in the case of rapid depth variations. This example shows that the 3D-model created with insufficient detail might not be reliable enough to take the real presumed advantage of the 3D information.

The method is implemented in Matlab, and the code has not been optimized for speed. It takes on average 6.87 seconds per frame on average, when executed on an Intel core 2 Duo 2 GHz processor. For initialization (step 1), it takes 134.6 seconds on average. However, being an online approach, the method has the potential to be implemented for real-time operation. Figure 3.1 shows the breakdown of the average execution time required for each step in the proposed method.

3. ONLINE SUNFLICKER REMOVAL USING DYNAMIC TEXTURE PREDICTION



(a) *Rocky* sequence



(b) *Andora* sequence

Figure 3.14: Comparison between the proposed and the offline method for the *Rocky* and *Andora* sequences

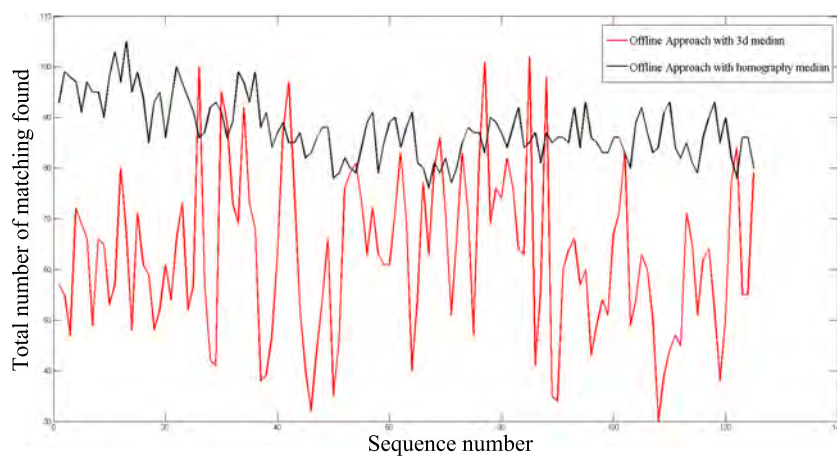


Figure 3.15: Performance comparison in the *Synthetic* sequences between the online and offline approaches using homography median in both cases.

3. Online Sunflicker Removal Using Dynamic Texture Prediction

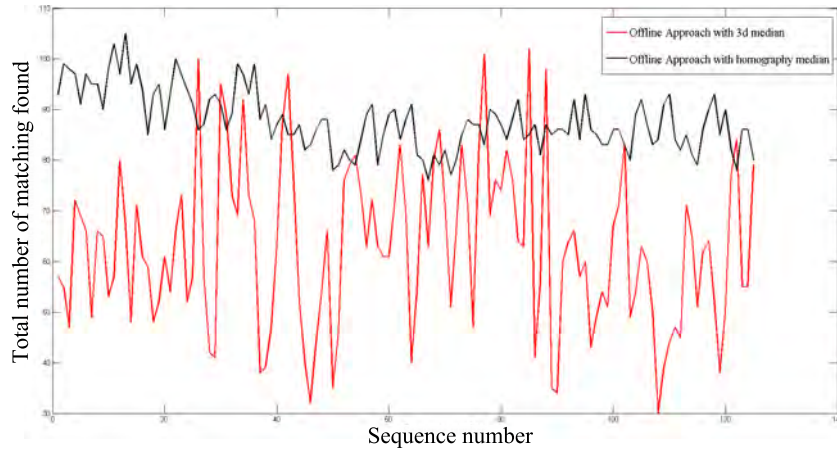


Figure 3.16: Performance comparison graph on the *Uneven* sequence

Table 3.1: Breakdown of the average running time required for each step in the proposed online sunflicker removal method.

| Step | Required time (in seconds) | Step | Required time (in seconds) |
|------|----------------------------|------|----------------------------|
| 1 | 134.64 | 6 | 2.10 |
| 2 | 0.01 | 7 | 1.29 |
| 3 | 2.78 | 8 | 0.23 |
| 4 | 0.34 | 9 | 0.01 |
| 5 | 0.11 | | |

3.8 Summary

Refracted sunlight generates dynamic patterns, which degrade the image quality and the information content of the acquired data. The development of an online method to partially or completely eliminate this effect is a prerequisite to ensure optimal performance of underwater imaging algorithms.

This work addresses the specific problem of sunflicker in shallow underwater images, by presenting a method suited for de-flickering on the fly. In the current online method, only the previous few frames with the current frame, are used to create the median image. In this case, the homography is calculated by registering the sunflicker removed version of the current image (using prediction from the dynamic texture model learned from the last few frames) with the last flicker free image. This results in higher image registration accuracy than in the offline method [37] where the registration is carried out over the original images affected by the illumination caustic patterns. The better registration results in better median image estimation and, ultimately, in better sunflicker correction.

An extension to the current work is to relate the illumination frequency with the number of frames required to perform the sunflicker removal. It can be a way to know beforehand the minimum frame rate of the camera required to remove the sunflicker effect properly. Also, for instrumented imaging platforms, the camera motion can be estimated using a motion sensor, such as a rate gyro or an accelerometer. This estimate can be used during the initialization phase of the method, or whenever image registration is not possible.

Chapter 4

Proposed configurable classification framework

This chapter describes the proposed classification framework for underwater object mapping using optical imagery. In the later part of this chapter, the implementation of our proposed method together with the set of used parameters are described. The parameters are tuned after experimenting with four benthic datasets. These tuned parameters can be directly used on any future underwater image dataset.

The proposed classification framework consists of six main steps, each having several sub-steps to perform the training and classification of underwater images. The proposed method performs a supervised classification under the assumption that enough labeled training data is available (at least 15 examples for the training of each class). A unique feature of our approach is that it is based on a configurable implementation scheme. Under this scheme, different options are available for each of the main processing steps. The configurable scheme allows for the tuning of the processing pipeline to match the characteristics of different datasets and leads to performance gain.

4.1 Introduction

The proposed classification framework contains six main steps where, for each step, several options, or sub-steps, are available. The steps and sub-steps of the proposed

4. PROPOSED CONFIGURABLE CLASSIFICATION FRAMEWORK

method are illustrated in Figure 4.1 and discussed below. The selection of each step and sub-step depends on the characteristics of the datasets to be classified. In Figure 4.1, the sub-steps colored in light blue are mandatory for all datasets. The light green sub-steps are optional. Grey colored sub-steps are mutually exclusive, meaning that only a single one in that particular step must be selected.

In this thesis, we have analyzed the performance of these sub-steps with respect to the characteristics of the datasets in order to choose the appropriate ones. The insights derived from the results can help configure the best lineup for future datasets. These insights derived are discussed further in Chapter 5.

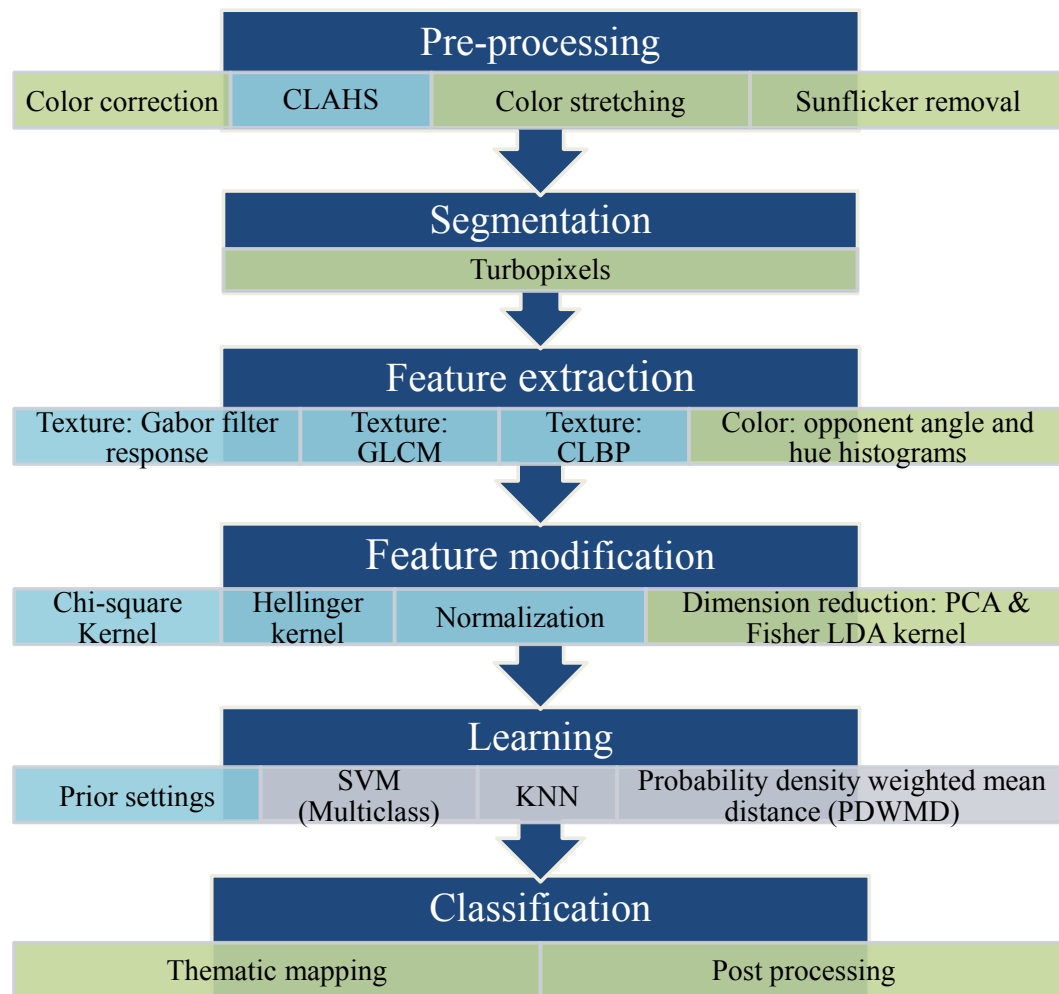


Figure 4.1: Proposed framework for supervised object classification using underwater optical imagery.

4.2 Overview of related methods

In this section, the theoretical background related to the sub-steps used in our proposed framework is discussed. In each of these steps and sub-steps, tools developed by the computer vision community are customized for underwater optical images.

4.2.1 Pre-processing

Images captured under non-uniform light conditions may lose information content due to color attenuation, scattering effects and low contrast. Schettini and Corchs [91] presented a summary of existing work on underwater image enhancement with the aim of retaining part of that lost content. Among the various image degradation aspects, the loss of contrast has a particularly strong negative impact in the classification. To enhance the contrast of the image and to consistently maintain image quality, we have adopted four different methodologies as pre-processing sub-steps as discussed below.

Color correction using known color references

Color correction using color difference optimization is possible for image datasets that have color markers in the image [2]. This sub-step is optional because it requires the presence of known color patterns in the image dataset, such as the two examples specified in Figure 4.2. In this example, there are nine color references available. However, we performed the color difference optimization with three color references. To work with color correction, a minimum of one color reference is required (preferably white).

Images from most cameras have a non-linear response to radiance. However, the color correction is more suited to being done in the linear domain [84]. This sub-step involves the conversion of the camera response to a linear domain, followed by a color correction and a conversion back to the original non-linear domain. The sub-step is described in the following.

The nonlinear to linear red-green-blue (RGB) transformation is performed based on the following equation. Here, R represents the nonlinear red, and r represents

4. PROPOSED METHOD

the linear red. The gamma value (γ), in general, varies from 2.2 to 2.4, depending on the camera's compression [84]. For JPEG image files, the gamma (γ) is normally 2.4. We used this value for the gamma in all the experiments in this thesis. The values of intensity clippings are 1.055 and 0.055. The same equation is applicable with different notations for the green and blue channels.

$$r = (((R/R_{\max}) + 0.055) / 1.055)^\gamma \quad (4.1)$$

The main goal in this sub-step is to find the color correction factor that converts the distorted color values to the original color values. Since the real color intensity of the markers is known, a linear multiplier for each channel of each image can be found, which converts the observed marker color to be as close as possible to the original color.

The coefficient for the red color channel modification is computed with the following equation:

$$C_r = \frac{\alpha_{r1}\tilde{r}_1r_1 + \alpha_{r2}\tilde{r}_2r_2 + \alpha_{r3}\tilde{r}_3r_3}{\alpha_{r1}r_1^2 + \alpha_{r2}r_2^2 + \alpha_{r3}r_3^2} \quad (4.2)$$

Here, \tilde{r}_1 is the ideal red component value of the first marker, r_1 is the red component of the first marker in the image, α_{r1} is the corresponding weight of the individual color channels and C_r is the red correction factor. In our case, we used the same weights for all three channels. However, the color red attenuates faster in the underwater environment.

The red correction factor C_r is multiplied with the linear version of the red color channel (r) to obtain an approximately corrected version r_f . r_f is the corrected version of r in the linear domain.

$$r_f = C_r \times r \quad (4.3)$$

4. Proposed method

The linear to nonlinear red-green-blue (RGB) transformation is obtained based on the following equation.

$$R_f = 0.5 + 255 \times (1.055 \times r_f^{1/\gamma}) - 0.055 \quad (4.4)$$

Figure 4.2 illustrates the results of applying color correction in 4.2(b) and 4.2(d).

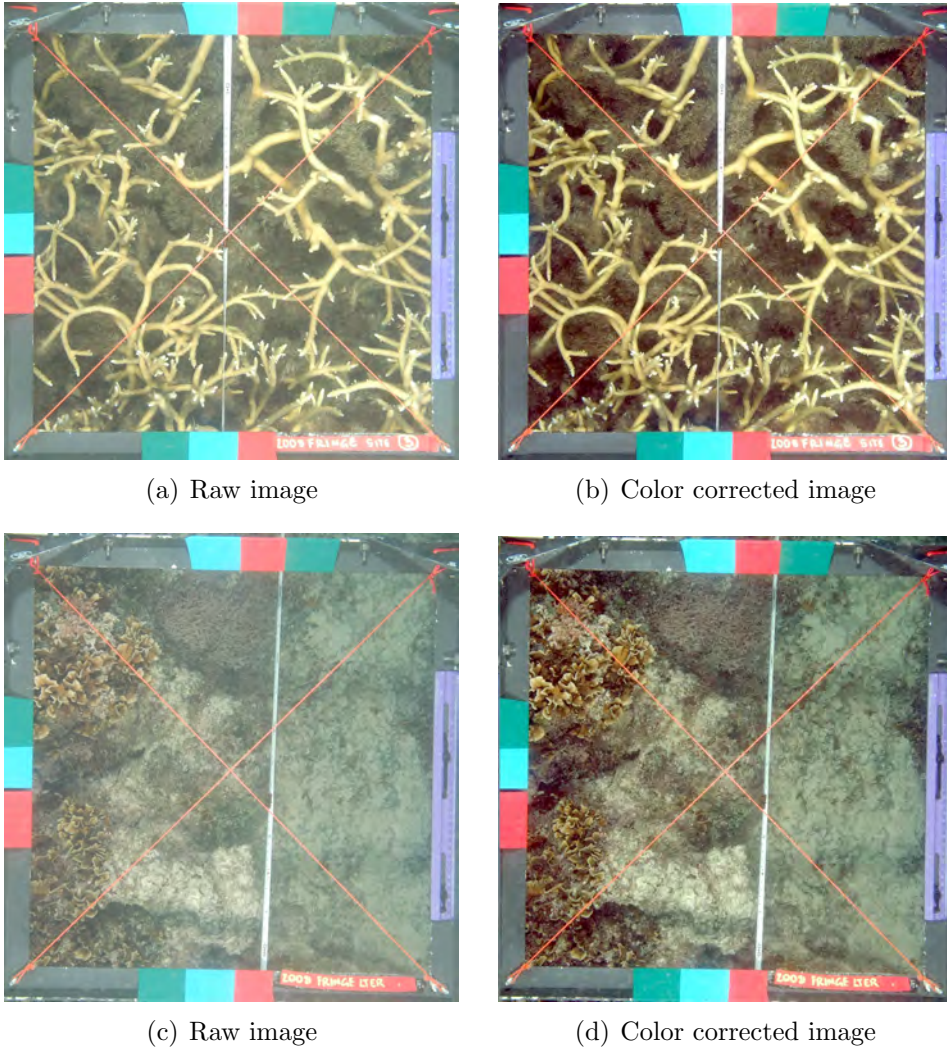


Figure 4.2: Illustration of the presence of color markers in raw images in the Moorea Coral Reef (MCR) dataset and color corrected images. There are three sets of color markers in these images. We used only one set from the top middle (comprising a three-color reference) to calculate the correction factors.

4. PROPOSED METHOD

Contrast limited adaptive histogram equalization (CLAHS)

The contrast limited adaptive histogram equalization (CLAHS) method was proposed by Zuiderveld *et al.* [112]. This method works better for low contrast non-uniformly illuminated underwater images in comparison to any other image enhancement method. This technique divides the image into equally sized non-overlapping regions. Then the histogram for each region is equalized using a monotonically non-decreasing grey level transform determined for that region. The transformation of the pixel values of each subregion is done so that the output histogram of any subregion approximately matches a specified histogram. In our implementation, a Rayleigh distribution with an alpha equal to 0.4 was used similarly to the implementation in [112]). A bilinear interpolation is performed across neighboring subregions to eliminate artificially induced boundaries.

The number of regions in which the image is subdivided is an important input parameter for this method. Figure 4.3 shows the effect of different numbers of subregions on an underwater lava image [33]. In Figure 4.3(d), it can be observed that a higher number of subregions provides an output with more uniform illumination. However, some of the important details are no longer distinguishable. This suggests that we may lose the grey level information of the texture by using an increasingly high number of subregions. The most suitable output can be seen in Figure 4.3(c), where the contrast has been sufficiently enhanced. Therefore, in our implementation we used 4×4 subregions. If the image is of a larger size, the number of subregions can be increased.

Color stretching

The contrast of the image can also be improved with color stretching. In order to perform color stretching for each individual channel, we determine the 1.5% and 98.5% intensity percentile, subtract the lowest one from all intensities in that channel and divide the result by the highest intensity. Figure 4.4 illustrates an example of color stretching on an image taken from the *MLC 2008* dataset (Appendix A).

4. Proposed method

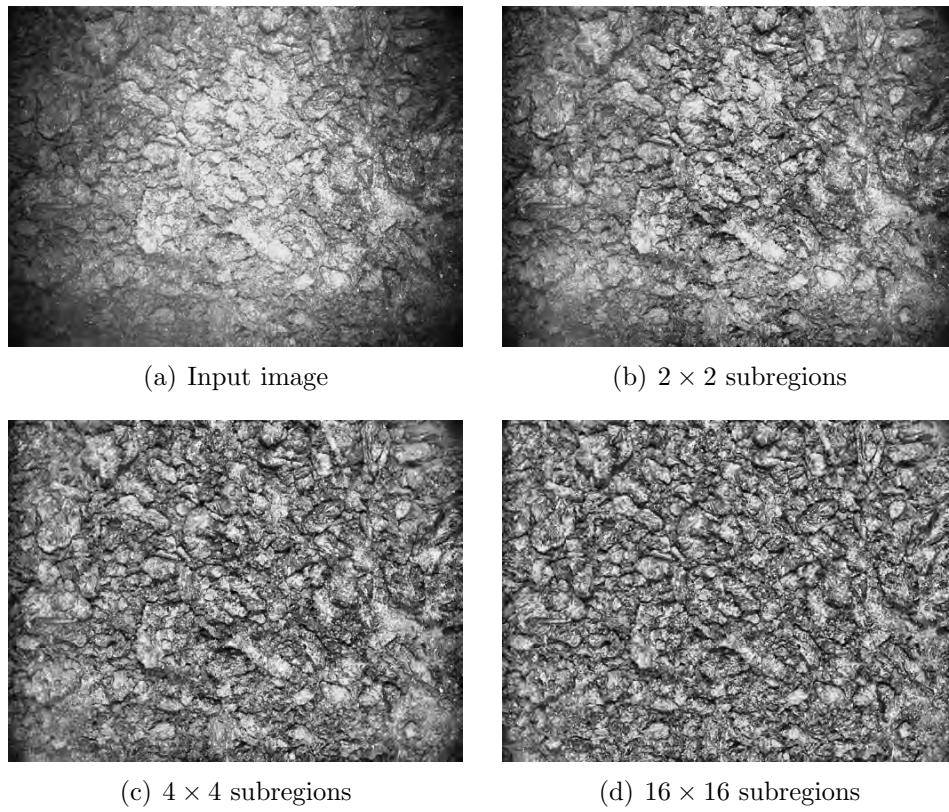


Figure 4.3: Output of CLAHS for different numbers of subregions: (a) Input image, (b) 2×2 , (c) 4×4 , (d) 16×16 .

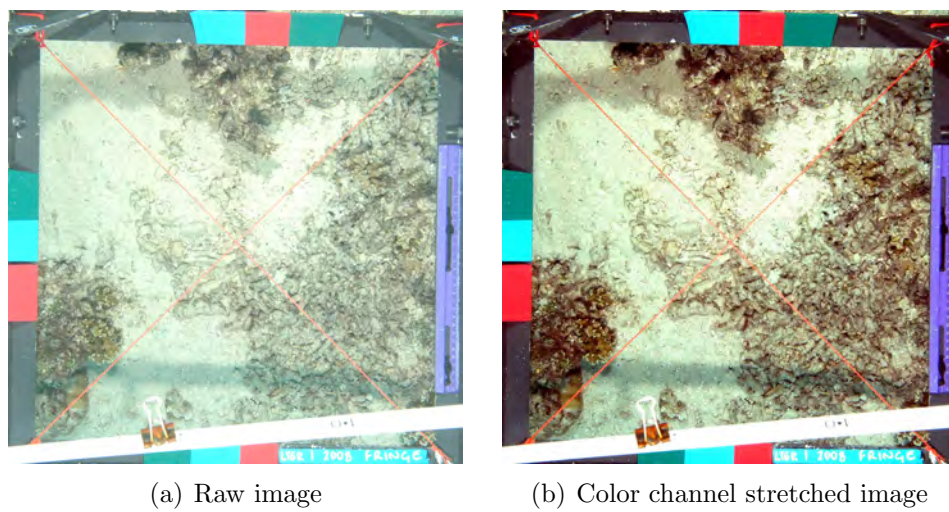


Figure 4.4: Example of the color stretching.

4. PROPOSED METHOD

Sunflicker removal

The method for sunflicker removal has already been presented in detail in Chapter 3. The images that have been strongly affected by sunflicker, can be improved in quality to help retrieve relevant information using a sunflicker removal method both offline [37] or online [94].

4.2.2 Segmentation

In the proposed method, we use a fast superpixel segmentation algorithm called 'TurboPixels' by Levinshtein *et al.* [57]. Superpixels are the result of a dense over-segmentation of an image into a lattice-like structure of compact regions that follow the local image boundaries. Superpixels reduce image resolution from a pixel level to a superpixel level through homogeneous pixel grouping while avoiding under-segmentation.

In the 'TurboPixels' method [57], initial seeds are randomly created and sparsely populated. The number of seeds is the same as the number of superpixels defined in the input parameter. Initial seeds are dilated using the geometric-flow-based algorithm (adapting to local image structure) [7; 51]. This method combines a curve evolution model for dilation with a skeletonization process on the background region to prevent the expanding seeds from merging. The algorithm applies region growing from the initial seeds, maintaining the following criteria for superpixels; uniform size and coverage, connectivity, compactness, smoothness, edge-preserving flow and no overlap. This method performs moderately well in terms of compactness, under-segmentation and boundary recall in comparison to other superpixel segmentation methods in the state of the art [92]. However, it is highly efficient in terms of speed. Figure 4.5 shows an example of superpixel segmentation using the TurboPixels method on a small part of the mosaic created from the North Sea survey.

One important parameter in the segmentation step of our method is the number of superpixels. The number of superpixels can be selected based on the amount of over-segmentation of the image desired, keeping in mind that if the number of superpixels increases, the resolution of the mapping and the classification accuracy increases; but the required time for computation increases as well. A comparative analysis of

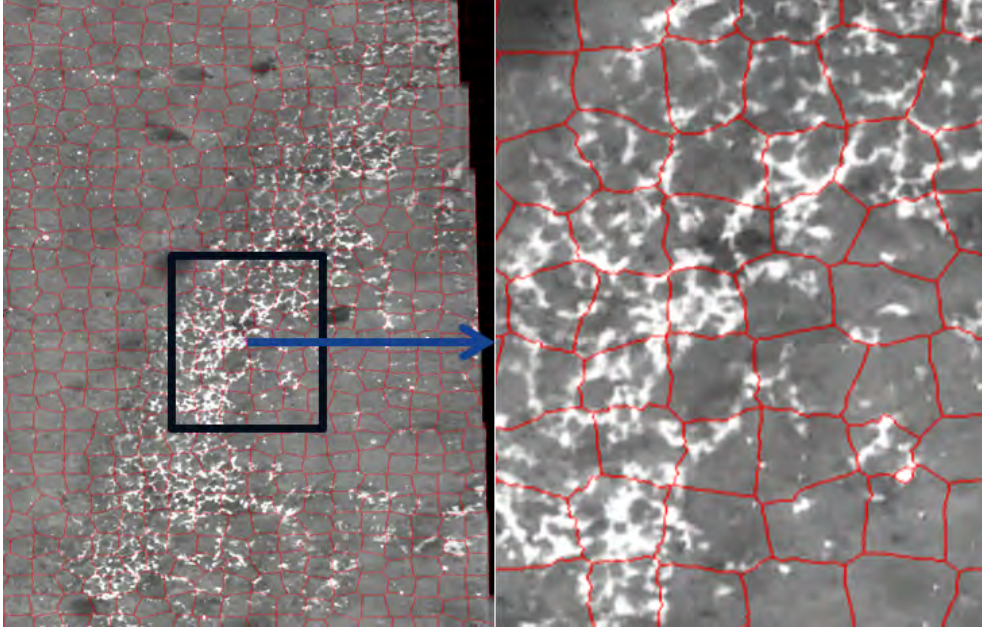


Figure 4.5: An example of superpixels segmentation using the TurboPixels method.

the classification accuracy and computational time based on a varying number of superpixels was performed to get more insight into this parameter. Experimentation was performed with a number of superpixels ranging from 281 to 8,000 created on the *North Sea mosaic* dataset (Appendix A). It was found that after a certain number of superpixels (approximately 3,000 in this case) the classification accuracy becomes stable but the time required increases linearly with the increase of the number of superpixels (as illustrated in Figure 4.6). In this case, each superpixel covers an area approximately $1/9$ the area of any patch in the training set. Therefore, for any new dataset, we can find the appropriate number of superpixels as below.

- Initially taking the superpixel area size as $1/9$ of the training patch size.
- Thereafter dividing the main image size by this superpixel area size to find the number of required superpixels.

4.2.3 Feature extraction

The texture of an object can carry important and reliable visual cues for applications such as object recognition, medical imaging, satellite imaging and remote sensing, among others. Due to the dominant presence of texture in underwater images of

4. PROPOSED METHOD

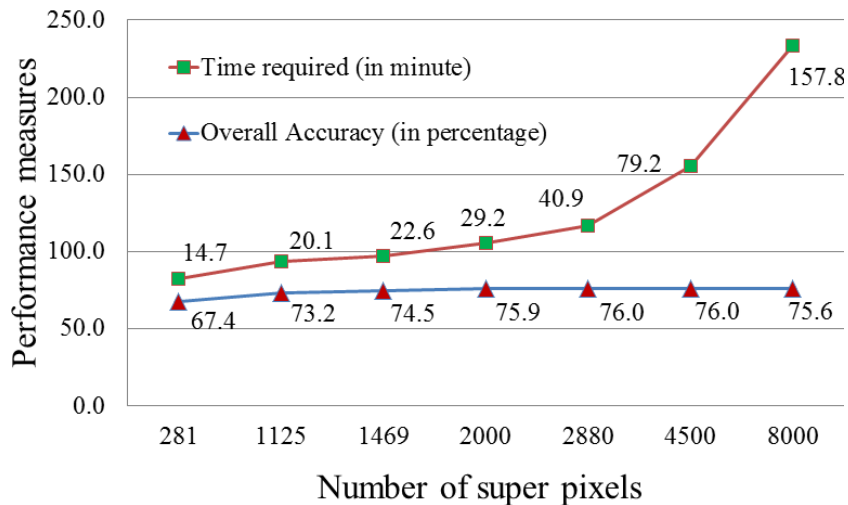


Figure 4.6: Effects on classification accuracy by varying the number of superpixels. Figure 4.6 illustrates one experimental result in terms of time and accuracy by varying the number of superpixels on *North sea mosaic* dataset described in Appendix A. In this experiment, the number of superpixels were varied, and for each different values the overall accuracy and time requirement for the superpixel creation were measured.

interest, texture information can be used to reliably solve the underwater object classification problem. To use texture features, it is necessary to define and represent the texture characteristics of each class as indiscriminately as possible. Also, the computational complexity for calculating these features is another important issue to consider. This chapter presents a succinct review of the state of the art methods of 2D-texture feature extraction methods considered in the proposed framework.

Gehler *et al.* [34] showed that a combination of several texture and color features leads to better image classification results than any single type of feature. In addition to texture features, for some classes, color features may contain discriminative information to help the classification task. Nonetheless, due to presence of color attenuation and illumination effects in underwater images, color information is not always reliable. The consistency of color must be ensured before using it as a feature. In the second part of this subsection, the hue and opponent angle color histograms are discussed as potential color features.

3D features can be useful of detection of objects (either man-made or natural) with particular shape or specific type of size or surface. In this work, we also proposed several 3D features and provided experimented results with *Ordnance dataset*. In the last part of this section, these 3D features are presented in details. As 3D

features can only be used for special types of datasets, they are not included in the main proposed framework.

Gabor filter response

Gabor filters are a group of 2D wavelets that take the form of a 2-D Gaussian modulated complex sinusoidal grating in the spatial domain [81]. Several applications on texture analysis using Gabor filter are shown in [5; 39; 42]. The general form of a Gabor filter is given in equation 4.5, where (x, y) are the coordinates of the filter, σ is the scale factor and (u_0, v_0) are the modulation frequencies along x and y directions respectively, which determine the center frequency $F = \sqrt{u_0^2 + v_0^2}$ and orientation $\theta = \arctan(\frac{v_0}{u_0})$.

$$h(x, y) = \frac{1}{2\pi\sigma} e^{-\frac{1}{2} \frac{x^2+y^2}{\sigma^2}} e^{-2\pi j(u_0x+v_0y)} \quad (4.5)$$

To obtain a representation of texture features, a set of Gabor filters with different scales and orientations is convolved with the image patch. Each Gabor filter generates a unique image response. The mean and variance of this image response are used as features for that particular Gabor filter. Figures 4.7 and 4.8 show examples of variation in Gabor wavelets in terms of scale and orientation. The number of scales and orientations of the Gabor filter have to be given as input parameters, as well as the center frequency of the wavelets to be created. In our implementation, by defining four scales and six orientations, we obtained 24 images of the filter response. Using the mean and standard deviation for each of these 24 images, a feature vector of 48 values is created.

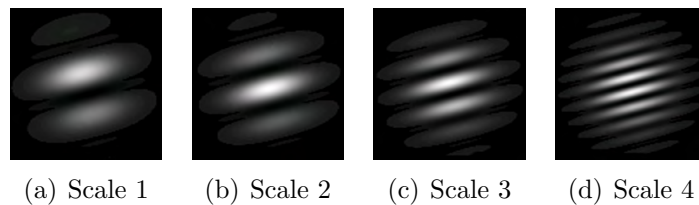


Figure 4.7: Examples of Gabor filter wavelet with scale variation.

4. PROPOSED METHOD

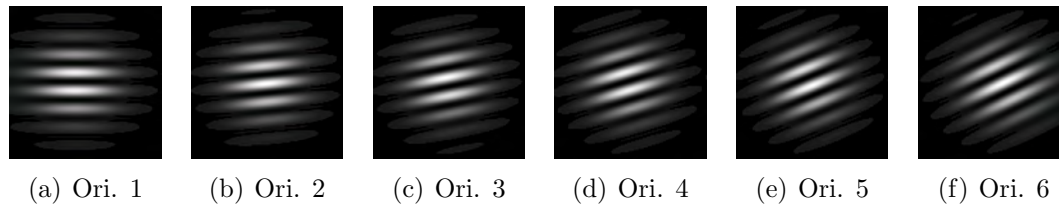


Figure 4.8: Examples of Gabor filter wavelet with orientation variation.

Grey Level Co-occurrence Matrix (GLCM)

Patterns on the regularity of occurrence of distinct grey levels in a texture image calculated using some statistics can be used as distinct texture features. A useful structure to present these patterns is the so called co-occurrence matrix [43], from which different statistics can be used to generate representative texture descriptors.

The co-occurrence matrix (CM) can be defined over an image as the distribution of co-occurring values at a given offset (d, θ) . The normalized co-occurrence matrix is represented as \mathbf{C} . I is the input image of size $n \times m$ pixels.

$$\mathbf{CM}(i, j) = \sum_{p=1}^n \sum_{q=1}^m \begin{cases} 1 & \text{if } I(p, q) = i \text{ and } I(p + \Delta p, q + \Delta q) = j \\ 0 & \text{otherwise} \end{cases} \quad (4.6)$$

$$\mathbf{C}(i, j) = \frac{\mathbf{CM}(i, j)}{\sum_{i, j} \mathbf{CM}(i, j)} \quad (4.7)$$

The number of occurrences of pixels within distance d and direction θ are tunable input parameters. The offset from center pixel can be presented as $[\Delta p \ \Delta q]$. We used $(0 \ 3)$, $(3 \ 3)$, $(3 \ 0)$, $(3 \ 3)$ offset values. These offset values represent 0, 45, 90 and 135 angular neighborhood degrees with a distance of three pixels from the center pixel. In our implementation, we calculated statistics individually for each offset and then took their average as final results.

The following list (Table 4.1) contains the indicators used to compute the co-occurrence matrix based features.. N is the number of distinct gray levels in a quantized image (16 in our case), μ , S and H are the mean, standard deviation and

entropy of C respectively.

Table 4.1: The list of indicators used for computing the co-occurrence matrix based features.

| |
|--|
| $\mu_x = \sum_{i,j} i \cdot \mathbf{C}(i, j)$ |
| $\mu_y = \sum_{i,j} j \cdot \mathbf{C}(i, j)$ |
| $S_x = - \sum_{i,j} (i - \mu_x)^2 \cdot \mathbf{C}(i, j)$ |
| $S_y = - \sum_{i,j} (j - \mu_y)^2 \cdot \mathbf{C}(i, j)$ |
| $\mathbf{C}_x(i) = \sum_j \mathbf{C}(i, j)$ |
| $\mathbf{C}_y(j) = \sum_i \mathbf{C}(i, j)$ |
| $\mathbf{C}_{x+y}(k) = \sum_{i=1}^k \mathbf{C}(i, k - i + 1)$ |
| $\mathbf{C}_{x-y}(k) = \sum_{i=1}^{N-K+1} \mathbf{C}(i, i + k - 1) + \mathbf{C}(i + k - 1, i)$ |
| $H_{xy} = \sum_{i,j} \mathbf{C}(i, j) \cdot \log \mathbf{C}(i, j)$ |
| $H_{xy1} = - \sum_{i,j} \mathbf{C}(i, j) \cdot \log(\mathbf{C}_x(i) \cdot \mathbf{C}_y(j))$ |
| $H_{xy2} = - \sum_{i,j} \mathbf{C}_x(i) \cdot \mathbf{C}_y(j) \cdot \log(\mathbf{C}_x(i) \cdot \mathbf{C}_y(j))$ |
| $H_x = - \sum_i \mathbf{C}_x(i) \cdot \log \mathbf{C}_x(i)$ |
| $H_y = - \sum_j \mathbf{C}_y(j) \cdot \log \mathbf{C}_y(j)$ |

The list of statistics used for GLCM feature calculations are given in Table 4.2. We have selected 22 features to be used in our method, following the work of Haralick *et al.* [43], Soh & Tsatsoulis [98] and Clausi [8]. We used these features as they are statistically independent and able to present the core texture characteristics of an object.

Completed Local Binary Patterns

The completed Local Binary Pattern (CLBP), proposed by Guo *et al.* [40], is a strong rotation invariant feature for texture classification. CLBP is an extension of the work by Ojala *et al.* [73] where they proposed the concept of a local binary pattern as an effective discriminative representation of texture characteristics. The flowchart of the CLBP feature generation method is illustrated in Figure 4.9 and described in the following.

4. PROPOSED METHOD

Table 4.2: The list of statistics used as GLCM features.

| | |
|---|---|
| Maximum Probability [98] | $\max\{\mathbf{C}(i, j) \forall (i, j)\}$ |
| Uniformity [43; 98] | $\sum_{i,j} \mathbf{C}(i, j)^2$ |
| Entropy [98] | $\sum_{i,j} \mathbf{C}(i, j) \cdot \log \mathbf{C}(i, j)$ |
| Dissimilarity [98] | $\sum_{i,j} \mathbf{C}(i, j) \cdot i - j $ |
| Contrast [43; 98] | $\sum_{i,j} \mathbf{C}(i, j) \cdot i - j ^2$ |
| Inverse Difference [98] | $\sum_{i,j} \frac{\mathbf{C}(i, j)}{1+ i-j }$ |
| Inverse Difference moment [98] | $\sum_{i,j} \frac{\mathbf{C}(i, j)}{1+ i-j ^2}$ |
| Correlation 1 [43; 98] | $\sum_{i,j} \frac{(i-\mu_x) \cdot (j-\mu_y) \cdot \mathbf{C}(i, j)}{S_x \cdot S_y}$ |
| Inverse Difference Normalized [8] | $\sum_{i,j} \frac{\mathbf{C}(i, j)}{1+ i-j /N}$ |
| Inverse Difference Moment Normalized [8] | $\sum_{i,j} \frac{\mathbf{C}(i, j)}{1+(i-j)^2/N^2}$ |
| Sum of Squares: Variance [43] | $\sum_{i,j} (i - \mu)^2 \cdot \mathbf{C}(i, j)$ |
| Sum Average [43] | $\sum_{i=1}^{2N-1} (i + 1) \cdot \mathbf{C}_{x+y}(i)$ |
| Sum Entropy [43] | $S_e = - \sum_{i=1}^{2N-1} \mathbf{C}_{x+y}(i) \cdot \log \mathbf{C}_{x+y}(i)$ |
| Sum Variance [43] | $\sum_{i=1}^{2N-1} (i + 1 - S_e)^2 \cdot \mathbf{C}_{x+y}(i)$ |
| Difference Variance [43] | $\sum_{i=1}^{2N-1} i^2 \cdot \mathbf{C}_{x-y}(i + 1)$ |
| Difference Entropy [43] | $- \sum_{i=1}^{2N-1} \mathbf{C}_{x-y}(i + 1) \cdot \log \mathbf{C}_{x-y}(i)$ |
| Information measure of correlation 1 [43] | $\frac{H_{xy} - H_{xy1}}{\max(H_x, H_y)}$ |
| Information measure of correlation 2 [43] | $(1 - e^{-2 \cdot (H_{xy2} - H_{xy})})^{0.5}$ |
| Auto-correlation [98] | $AC = \sum_{i,j} i \cdot j \cdot \mathbf{C}(i, j)$ |
| Correlation 2 [43; 98] | $\frac{(AC - \mu_x \cdot \mu_y)}{S_x \cdot S_y}$ |
| Cluster Shade [98] | $\sum_{i,j} (i + j - \mu_x - \mu_y)^3 \cdot \mathbf{C}(i, j)$ |
| Cluster Prominence [98] | $\sum_{i,j} (i + j - \mu_x - \mu_y)^4 \cdot \mathbf{C}(i, j)$ |

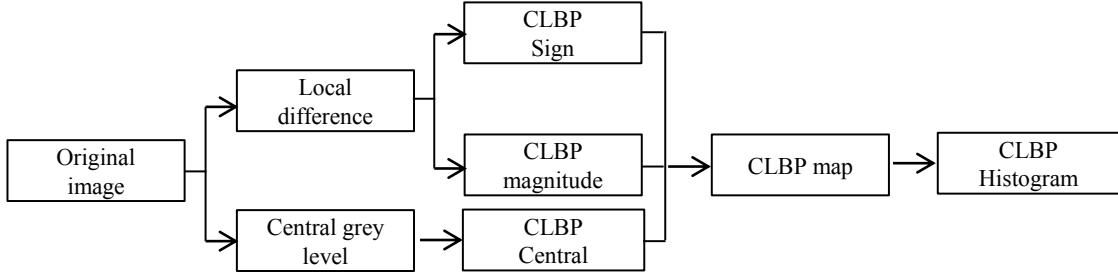


Figure 4.9: Flowchart of Completed Local Binary Pattern (CLBP) feature generation.

- A local binary pattern is created by comparing the center pixel of the sliding window with the neighboring pixels. The number of neighbors, which depends on the size of the sliding window, can vary depending on the distance set from the center pixel. As illustrated in Figure 4.10, g_c is the center pixel. It is possible to generate evenly spaced points g_p from the center pixel, g_c at P radius where $p = 0, 1, \dots, P - 1$. The difference between of g_p and g_c can be presented as a vector $[d_p, \dots, d_{p-1}]$ called the difference vector. The individual difference is calculated using $d_p = g_p - g_c$.

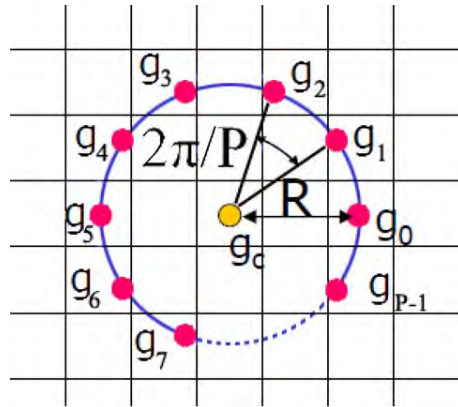


Figure 4.10: Example of a Completed Local Binary Pattern (CLBP) creation with a center pixel; its P circularly & evenly spaced neighbors with radius R .

- The difference between g_p and g_c (represented as d_p) can be further broken down into two components; sign and magnitude using equation 4.8. The sign reflects if the value of the neighboring pixel is higher (positive sign) or lower (negative sign) than the center pixel. The neighbors with zero difference from the center pixel are considered as a positive sign. The magnitude stores the absolute intensity difference of the neighboring pixel from the center pixel. Figure 4.11 illustrates an example of a sample block broken down into sign and

4. PROPOSED METHOD

magnitude. The signs can be directly used as binary bits. The magnitudes are compared with a threshold value (the mean value of magnitudes) to convert it to binary bits. The center pixel intensity value is the third component that is also converted to a binary bit by comparing with a threshold (average intensity).

$$d_p = s_p * m_p \text{ where, } s_p = \text{sign}(d_p); m_p = |d_p| \quad (4.8)$$

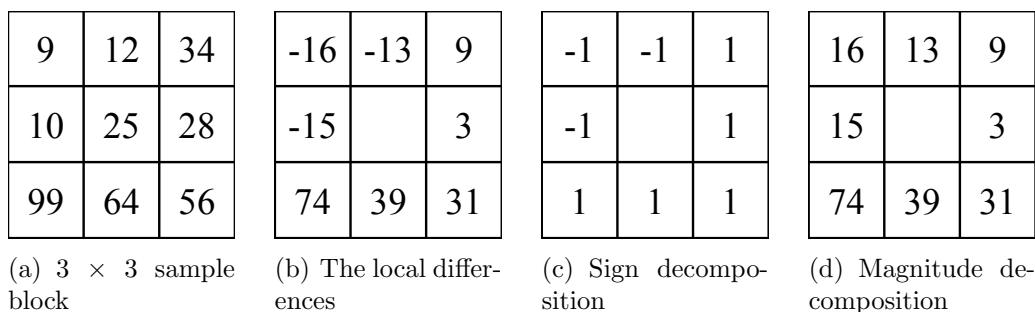


Figure 4.11: Example of Completed Local Binary Pattern (CLBP) generation.

- Using these three operators (sign, magnitude and center pixel intensity), all possible patterns are generated as bins for the CLBP histogram. All the similar patterns that are similar apart from those shifted by rotation are counted into a single bin in the histogram to include rotation invariance. Also, all the less probable patterns [74] are counted as a single bin. In this way, the feature vector stays compact and more discriminative. Ojala *et al.* [74] suggested that the majority of local binary patterns (LBP), which is the sign portion of the CLBP, in a texture have a limited (i.e. maximum two) number of transitions between zero and one (termed 'Uniform'; e.g. 00001111 is uniform but 01010101 is non-uniform) and correspond to micro-features such as edges, spots and corners. 'Non-uniform' patterns usually correspond to a small proportion of patterns and cannot be estimated individually, therefore, all non-uniform patterns are counted as a single histogram bin.
- For implementation, all the pixels in the image patch are analyzed for CLBP patterns. The found patterns are counted in the corresponding bins in the CLBP histogram. The final counts of the histogram bins are normalized to

create the CLBP feature vector of the input image. In our implementation, we use the rotation invariant uniform format resulting in a histogram of 108 bins when concatenated for three window sizes of 8 (20 bins), 16 (36 bins) and 24 (52 bins) pixels.

There are some additional works available in the literature as possible extensions to CLBP features. For example, Liao *et al.* [58] suggested the use of "Dominant Local Binary Patterns" (DLBP). According to them, uniform patterns are suitable for textures that contain mostly straight or low curvature edges. Textures tend to have a significant number of non-uniform patterns if they are highly irregular and have complicated shapes, high curvature edges, crossing boundaries or corners. Such complicated and irregular patterns are better learned by the DLBP patterns. These DLBP patterns are found using a statistical approach on several standard texture datasets.

Hue and opponent angle histograms

Color features used for classification are intended to contain the following core properties.

1. Robustness to photometric variations such as shadow, shading, specularities and changes of light sources.
2. Robustness with respect to geometrical changes such as viewpoint, zoom, and object orientation variations.

The work by Finlayson *et al.* [25] and Weijer and Schmid [107] tried to achieve these properties in their proposed color features. In our work, we used the opponent angle and hue histograms proposed by Weijer and Schmid [107]. Their method performed better than the Finlayson *et al.* method [25] based on comprehensive color image normalization in terms of classification accuracy. The hue (H) and opponent angle (ψ_0) are color channels derived from red (R), green (G) and blue (B) color channels using the following equations. Here ' dim ' denote special coordinates (x, y); and R_{dim} , G_{dim} , B_{dim} denotes the first order derivatives of (R,G,B) with respect to dim .

4. PROPOSED METHOD

$$H = \frac{\sqrt{6}*(R-G)}{\sqrt{2}*(R+G-2B)} \quad \psi_0 = \frac{\sqrt{6}*(R_{\text{dim}}-G_{\text{dim}})}{\sqrt{2}*(R_{\text{dim}}+G_{\text{dim}}-2B_{\text{dim}})} \quad (4.9)$$

These two newly derived channels (hue and opponent angle) are more robust to photometric variations than the raw (R,G,B) channels [24; 106]. We have modified the formula used in their work by removing the arctan in both the equations. These modified version works faster providing relatively the same results. In our implementation, two separate histograms are created as color descriptors using these two channels. The number of bins in the histograms is an input parameter. We are using 36 bins per color histogram following the implementation in the Weijer and Schmid method [107]. We use color features only in case the colors of the object in the images appear reliably as a discriminative class identity.

3D features

For both natural and man-made objects, 3D or 2.5D cues can provide important discriminative information for classification. 3D or 2.5D features can work together with 2D features as additional descriptors to identify objects.

When there is enough overlaps among the images used to create the mosaic, a 3D reconstruction of the seabed can be created. From the constructed 3D model, an elevation map (depth map) of the mosaic can be generated. From the elevation map several 3D features can be calculated, as described in the following section. An elevation map extracted from a 3D model contains all the information needed to calculate the 2.5D features.

Our approach to create 3D textured models of the seabed using structure from motion (SfM) method from a moving platform is an extension of existing 2D (Gracias et al. [38]; Lirman et al. [60]) and 3D mosaicing algorithms (Nicosevici et al. [72]). The estimation of 3D structure starts by identifying common points across pairs of images using a variant of the SIFT [62]. Next, a robust sampling technique is used to identify and eliminate false point matches among images, resulting in multiple image observations of the same set of 3D points [30]. Finally, sparse bundle adjustment (Triggs et al. [103]) is used to obtain a set of 3D points that best comply with the image observations. Figure 4.12 illustrates examples of 3D reconstruction phases of

a portion of the *Ordnance* dataset.

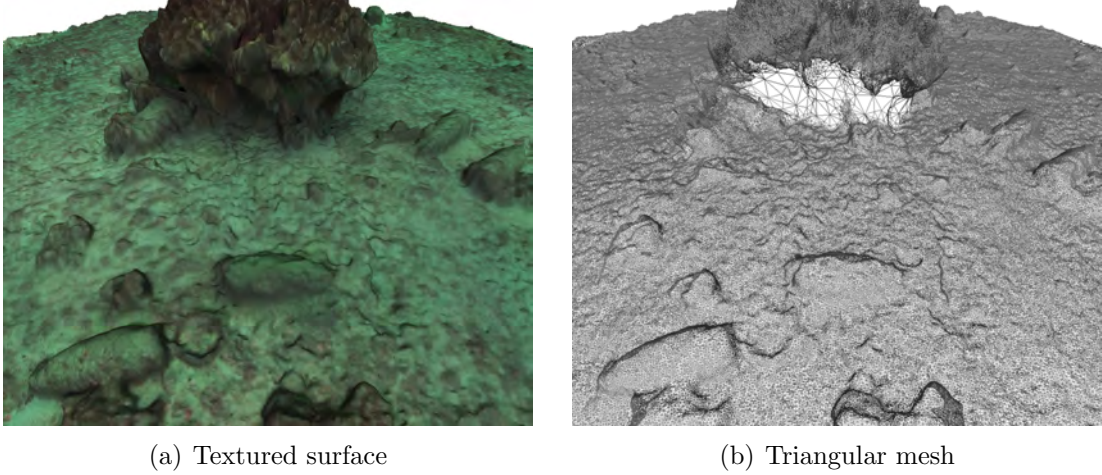


Figure 4.12: 3D reconstruction of a portion of the *Ordnance* dataset displayed as a textured surface (a) and as a triangular mesh (b). The scale varies in this oblique view with the lower (closer) edge of the image corresponding to 2 m on the seabed.

An important problem that affects the creation of 3D models is the large amount of data and computational processing involved. To address this issue, we used a combination of parallel processing toolboxes available for the Matlab programming language as well as libraries that allow the use of general purpose graphics processing units (GPGPU) to accelerate some of the highly parallelizable tasks, such as the feature extraction and dense point matching (Sinha et al. [97]).

The 2.5D features are calculated from an elevation map of the surface which is created from the 3D model of the surface. In order to create an elevation map a fixed reference plane is needed, where we can project the unconstrained geometry of the reconstructed 3D surface orthogonally. The center of the plane is found using the centroid of the vertices with a least square fitting. The normal of this reference plane is computed using PCA regression. Starting with the covariance matrix constructed from the vertices of the surface triangle mesh, we estimate the normal by computing its 3rd smallest eigenvector.

For each vertex in the surface, we can now define a height value with respect to this initial plane. Using all these height values, we fix our reference plane to pass through the one having smallest height, i.e., we move the of the previously computed plane to coincide with that of the lowest vertex in the set. In this plane, we want to sample an image using a regular grid of the desired size and resolution. For each

4. PROPOSED METHOD

of these discrete positions to sample, we generate a ray passing through the sample point, and perpendicular to the plane.

Due to the large number of queries, we speed up the ray-surface intersection test using an Axis-Aligned Bounding Boxes (AABB) tree structure [79]. The AABB tree component can provide a static data structure which makes it easier to perform efficient intersection and distance queries on sets of finite 3D geometric objects. Now, having an intersection point for each grid sample, the height is computed giving a value to the sample using the distance from the reference plane. Note that the ray intersection queries are computed from top-to-bottom, while the computation of the height value is bottom-to-top. Figure 4.13 illustrates the steps of creating elevation map.

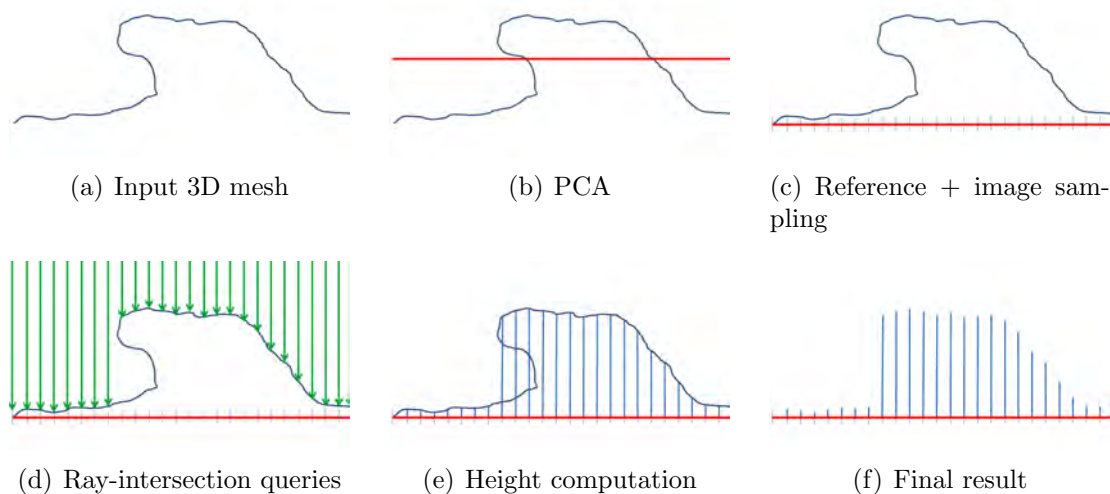
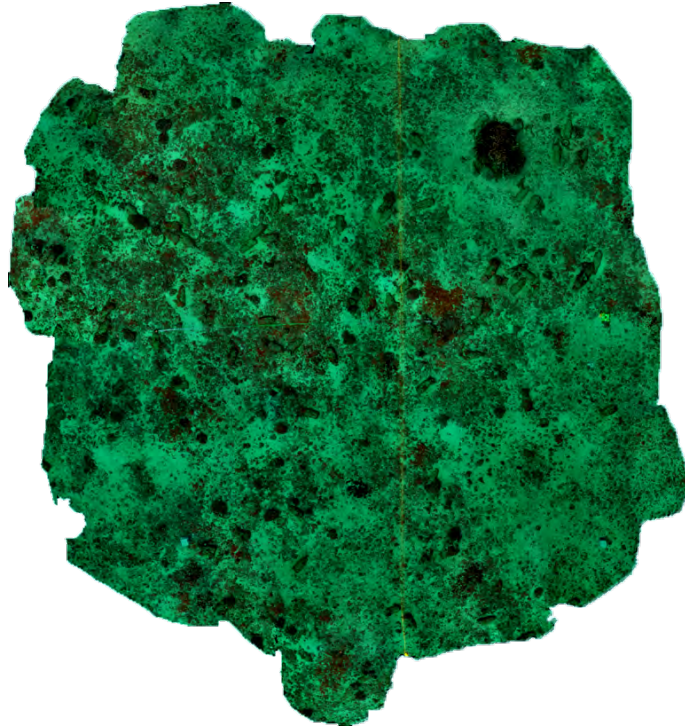


Figure 4.13: Step by step illustration of the process of creating elevation map.

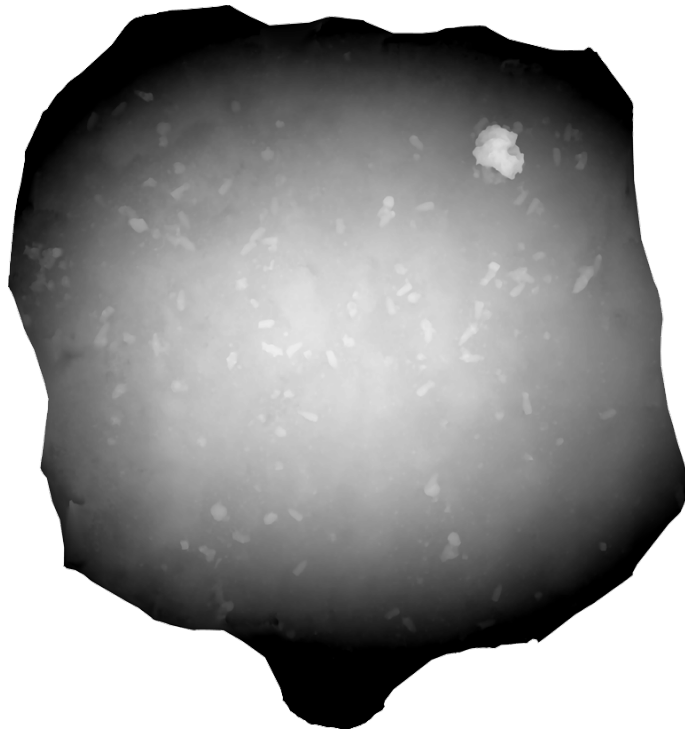
Figure 4.14 illustrates an example of a mosaic image with corresponding elevation map image created from *Ordnance dataset* (Appendix A). The following features were extracted from the elevation map of the patch, and used to generate the 3D feature vector.

1. Coefficients of polynomial surface fitting [49]

The elevation map of a patch can be approximated by a 2.5D polynomial surface modeled by a polynomial equation [4.10] of second degree [49]. The coefficients of the polynomials are extracted through least square fitting and



(a) Ordnance mosaic



(b) Elevation map of Ordnance mosaic

Figure 4.14: Example of mosaic image with corresponding elevation map created from *Ordnance dataset*.

4. PROPOSED METHOD

used as features. Each polynomial in each patch provides nine 2.5D features. Figure 4.15 shows the example of an estimated polynomial surface of a patch from *Ordnance* dataset.

$$f = p_1 + p_2x + p_3y + p_4x^2 + p_5xy + p_6y^2 + p_7x^2y + p_8xy^2 + p_9y^3 \quad (4.10)$$

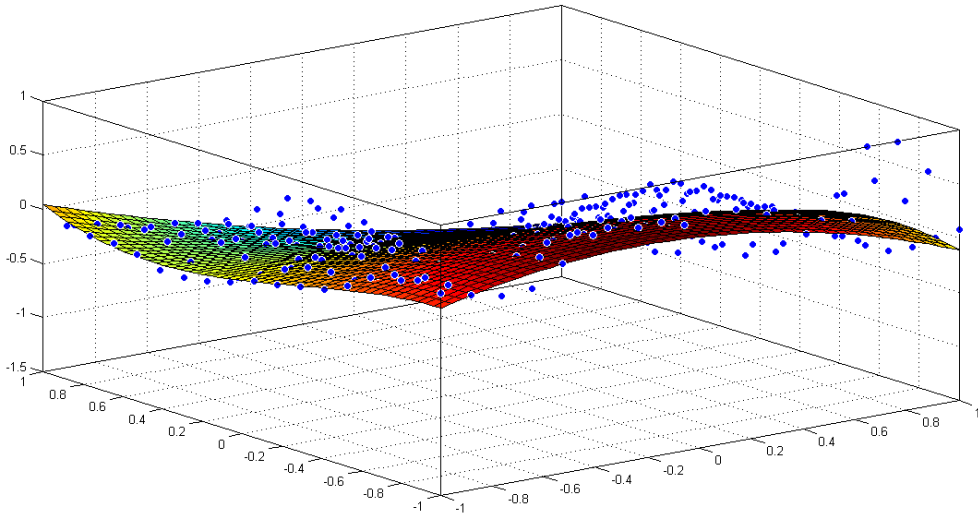


Figure 4.15: Example of polynomial surface fitting of a patch from *Ordnance dataset* using sparse points.

2. Standard deviation, skewness and kurtosis of elevation [54]

From the elevation map we calculate the standard deviation, skewness and kurtosis using the altitude values of each pixel in the patch. Thereby, three more 2.5D features are extracted. For calculating skewness and kurtosis, the following two equations are used. Here Z_i represents the height of a 3D point, Z_m and S are the mean and the standard deviation of the 3D points, N is the number of data points.

$$S = \frac{\sum_{i=1}^N (Z_i - Z_m)^2}{N} \quad (4.11)$$

$$Skewness = \frac{\sum_{i=1}^N (Z_i - Z_m)^3}{(N-1)S^3} \quad (4.12)$$

$$Kurtosis = \frac{\sum_{i=1}^N (Z_i - Z_m)^4}{(N-1)S^4} \quad (4.13)$$

3. Angle of the principal patch plane [30]

We fit a 3D plane as the principal plane of the patch. The principal plane is calculated from a least squares solution. The angle of this 3D plane with the ground plane is considered as another 2.5D feature.

4. Mean and standard deviation of the distance of the 3D points to the principal plane [30]

Two more 2.5D features are given by the mean and standard deviation of the distance of the points on the elevation map to the principal plane computed in point 3. These distances are measured from the points on the elevation map to the corresponding projected points on the principal plane of the surface.

5. Rugosity [30]

Rugosity is commonly used in marine science to characterize seafloor habitats. Rugosity can be defined as a small scale variation of amplitude in the height of a surface, and is considered as an important 2.5D feature [64]. Rugosity, r of a patch is calculated by dividing the area (A_s) of the contoured surface bounded by the patch, by the area (A_p) of the orthogonal projection of the surface onto the principal patch plane. Figure 4.16 illustrates an example of surface area and area on the principal plane of the patch. The area of the surface (A_s) was created by the Friedmans method [30]. In this method, the surface is considered as a set of triangles created by sampling points. By calculating the area of each triangles and summing up all, the total area of the surface is calculated. For the principal plane, it can be considered as 3D rectangle where four vertices are the four points at each corner of the patch. Using these four 3D points, the area of the principal plane of the patch is calculated. This principal plane decouples the slope effect, which otherwise might result in a higher rugosity value on a slope containing a relatively smooth surface.

4. PROPOSED METHOD

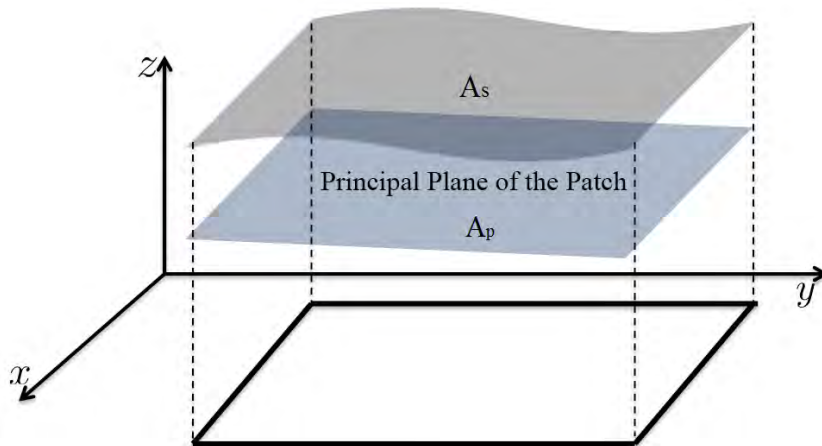


Figure 4.16: Example of surface area (A_s) and area on the principal plane (A_p) of the patch.

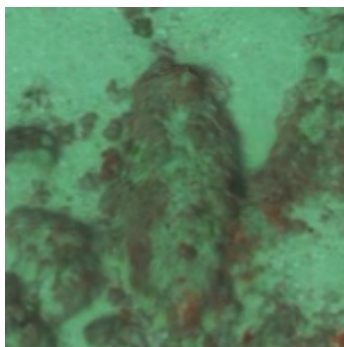
$$Rugosity = \frac{A_s}{A_p} \quad (4.14)$$

6. Symmetry [54]

Symmetry is an effective method to identify the structure of objects specially man-made and some specific natural habitats. Man-made objects usually represents higher recognizable symmetry, or partial symmetries. Generating a symmetry image of a patch and then taking the mean and variance of that image can produce two additional 2.5D features from the optical mosaic or the elevation map. The symmetry image was created using the method by Kovese [54]. In this method, a Log Gabor filter [18] is used to measure the phase symmetry at each pixel of the image patch.

Figure 4.17 illustrates the effect of applying the phase symmetry on elevation map. In this figure it can be seen that the phase symmetry is differentiable among bombshell and background when applied on the elevation map image created from the *Ordnance* dataset. This figure illustrates that phase symmetry from the elevation map image can be considered as a relevant 2.5D feature since it encodes important shape information.

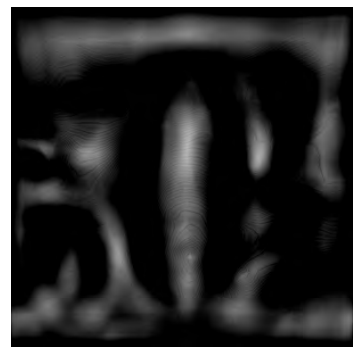
Finally, combining all these 2.5D features, we can extract a feature vector of 19 components. These features can be used individually or concatenated with the 2D



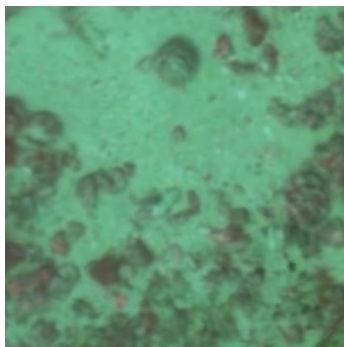
(a) Input image containing bombshell



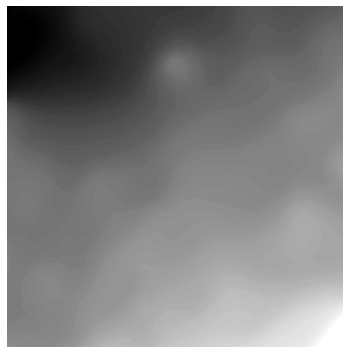
(b) Corresponding elevation map



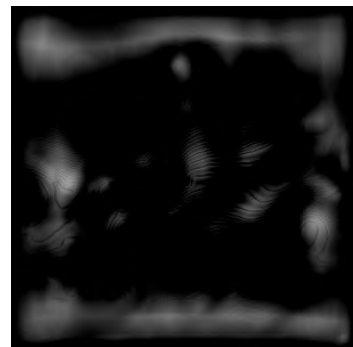
(c) Phase symmetry image



(d) Input image containing background



(e) Corresponding elevation map



(f) Phase symmetry image

Figure 4.17: Examples of phase symmetry images from corresponding elevation maps containing bombshells and background respectively.

4. PROPOSED METHOD

features for the classification purpose.

4.2.4 Feature modification

The calculated features using different methods capture the core properties of the object class and are more discriminative than the raw image patch. However, these features can be normalized and are able to be projected on more compact subspace. In the following subsection, the kernel mapping, dimension reduction and normalization methods used in our method are described.

Kernel mapping

Kernel mapping is used to project the feature vectors onto a linearly separable feature space. For the histogram features (in our case, the CLBP, opponent angle and hue histograms), the rarest bins tend to contain more discriminative information, since the most frequent bins correspond to the background pixels. The chi-square and Hellinger kernels (also known as the Bhattacharyya coefficient [50]) emphasize the low frequency bins, as illustrated in Figure 4.18, thus promoting the discriminative power. We use chi-square and Hellinger kernel mappings [98] to modify the features as shown in equations 4.15 and 4.16 respectively. Here, h and h' are normalized histograms, where h' is obtained from h using first order differentiation. k is the kernel function, γ is a regularization coefficient, and i and j are the indices of the histogram bins.

- Chi-square kernel function [98]

$$k(h, h') = \exp \left(-\frac{1}{\gamma} \sum_j \frac{(h_j - h'_j)^2}{h_j + h'_j} \right) \quad (4.15)$$

- Hellinger kernel function [98]

$$k(h, h') = \sum_i \sqrt{h(i) \times h'(i)} \quad (4.16)$$

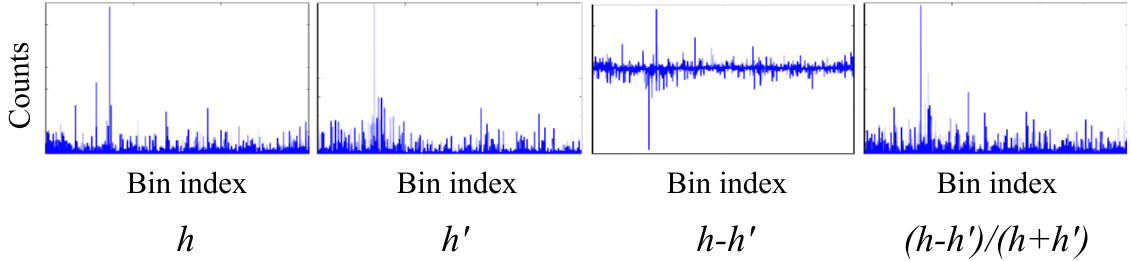


Figure 4.18: Example of variations in histogram bin counts after applying Chi-square kernel mapping. The fourth plot is the result of applying Chi-square kernel mapping on the histogram, which corresponds to the first plot. If we look at the last image and compare it with the first, we can see, the difference between the lower bin counts and the higher bin counts are reduced. This is more clearly visible in the right part of these images.

Normalization

Normalization is used to rescale all the features so that they are comparable. Before concatenating different independent features, a normalization of these features is carried out. The feature vector matrix is of size $N \times D$ where N is the number of samples and D is the number of features. The feature matrix is extracted from the training samples and the normalization is applied over it. In this matrix, each column can be normalized individually using equation 4.17. Here f_{nk} is the element of \mathbf{F} in line n and column k . f_{kmin} and f_{kmax} are the minimum and maximum values respectively in column k of the feature vector matrix. $f_{norm_{nk}}$ is the normalized value of f_{nk} .

$$f_{norm_{nk}} = \left(\left(\frac{f_{nk} - f_{kmin}}{f_{kmax} - f_{kmin}} \right) - 0.5 \right)^2 \quad (4.17)$$

Dimension reduction

Redundant or statistically correlated features may adversely affect the decision boundaries generated by any classifier. Feature dimension optimization can improve the orthogonality as well as the compactness of the feature space representation. Principal Component Analysis [111] is a popular way of removing data redundancy, however, it is not always the best solution in terms of increasing discriminatory among features. Linear discriminant analysis (LDA) [111] on the other hand, has proven to provide a higher discriminative power than PCA for feature reduction.

4. PROPOSED METHOD

LDA aims to maximize the ratio of scatter distributions between the class groups and individual classes. Compared to PCA, LDA also attains better generalization and less likeliness to over-fitting. However, it requires the computation of the inverse of the inner class scatter matrix, which may cause computation instabilities. Computation instabilities may happen if the matrix is close to being singular due to a low number of samples. One efficient way for feature reduction would be to project data onto the PCA space to remove the singularity and then project the PCA space into the LDA space, providing higher discrimination. Additionally, the number of dimensions selected for the PCA subspace affects the system’s classification performance, therefore, a suitable number of dimensions should be sought experimentally. For the experimental work in this thesis, an exhaustive search of all possible dimensions (less than the initial feature dimension) was performed to find the optimum dimension size for the individual datasets where the classification accuracy is maximum. This exhaustive search was done in the training step over validation data. Thus, the found dimension size can be used directly on the testing data. This exhaustive search is very time expensive.

Datasets, where classes have high inter-class variability and low intra-class variability, are considered as highly correlated. For highly correlated datasets such as *EILAT*, *RSMAS*, *EILAT 2*, *BMAT*, *CURET*, *KTH-TIPS* and *UIUC* (Appendix A), it was found that reducing the dimension of the feature space using PCA with Fisher kernel mapping (LDA) [70] increases the accuracy. This sub-step is optional, as it is mainly used for reducing the computational requirements in the final classification stage.

4.2.5 Learning

In this section, we describe the three different learning options (mutually exclusive) used in our methods. Whenever available, a prior probability of the test classes is used as additional information. When this prior is not available, it can be estimated from the training data. In Appendix A, the datasets used in this work are described in details.

The probability (prior) of any image patch falling into any possible class can be approximated in two ways:

- Class frequency: an estimate of the class probability calculated from the class frequency within the training data. Class frequency can bias the classification result, therefore it may only be used when the probability of occurrence of some classes are much higher than the others;
- Equal probability: assuming that an image patch has an equal probability of falling into any classes.

Estimates of the prior probability can help the classification of datasets especially in cases where it is difficult to get very accurate class boundaries. For underwater images, the use of estimated class frequencies tends to produce better results than assuming an equal probability, because in a natural scenario the classes are seldom evenly populated [86]. For the experimental part, any of the *K nearest neighbors* (*KNN*), the *Probability density weighted mean distance* (*PDWMD*) and the *support vector machines* (*SVM*) classifier are used depending on the characteristics of the dataset (explained in detail in Chapter 5).

K nearest neighbor

K Nearest Neighbor (*KNN*) [14] is a well known classifier frequently used in various applications by the computer vision community. This method is easy to implement and is based on computing the Euclidean distances of the test sample to all the training samples. The main reasons behind the popularity of the *KNN* algorithm are that it produces results with accuracy close to the state-of-the-art methods, it has strong consistency (even if the amount of test data is very large), and that it guarantees to yield an error rate not worse than twice the Bayes error rate (the minimum achievable error rate given the distribution of the data) [75]. However, for large datasets (for example, datasets with more than 12,000 samples), the effectiveness with respect to classification accuracy of this method is negligible, owing to the higher storage requirements, lower efficiency in classification response and lower noise tolerance. Although the *KNN* algorithm was originally proposed several decades ago, improvements are still being published, such as the work of Toussaint *et al.* [102]. Several recent works have proposed ways to improve the computational cost of the *KNN* algorithm [75]. In our implementation, we have used the standard formulation.

4. PROPOSED METHOD

Probability density weighted mean distance

The *probability density weighted mean distance* (PDWMD) algorithm was proposed by Stokes and Deane [100] for classification. In this method, every patches are evaluated by computing the Euclidean distances between the considered patch and all the training patches for one class. In this way, each of the classes is considered in turn. The mean of the three smallest distances for one class is taken to be the final distance for that class. The final label is determined by choosing the class with the smallest mean distance. This method attained a good level of consistency in terms of classification accuracy for most of the underwater image datasets as shown in Chapter 5.

Support vector machines

Support vector machines (SVM) by Cortes and Vapnik [12] is one of the most commonly used classifiers. This method can achieve state of the art accuracy with a relatively low execution time. The main advantages of SVM are that it ensures good generalization, and the learning involves optimization of a convex problem [12]. The performance of the method greatly depends on the choice of input parameters such as the cost function, the kernel function, the binary or multiclass (one to one or one to all) classification scheme and the normalization (L1 or L2 norm) type. For individual datasets, appropriate parameters can be derived experimentally. In our work, we use linear SVM with multiclass one to all schemes.

4.2.6 Classification

Once the learning is performed, any new data maybe be classified. When classifying patch by patch on a large image, additional information of spatial coherence or neighborhood consistency can and should be used for post processing the classification result. This step can help remove the scattered misclassifications on large images, and thereby improve the classification accuracy.

Thematic mapping

Thematic mapping is the final step in the proposed framework to create a segmented and classified map of an image that is visually interpretable. The thematic mapping sub-step of our approach addresses the classification of large area optical maps of the seabed. These large area representations (commonly known as image mosaics) are obtained by registering and blending large sets of individual images acquired at close range [23; 36; 59; 60]. For thematic mapping, these large images are segmented sequentially into superpixels and each superpixel is classified individually. For classification, patches around the center of each superpixel in the large input image (or mosaic) is taken and classified using the learning methods described earlier. Thereafter, all the classified superpixels are rechecked using neighborhood consistency as discussed in the following sub-section.

Post processing

While creating the thematic map, each superpixel or patch is classified with a confidence level depending on the distance from the class boundary. Class label is assigned if the confidence exceeds a user defined confidence threshold. The confidence threshold is chosen to a value so that 90% of the confidence values stay above it. In cases where the confidence is less than the confidence threshold, the corresponding superpixels or patches are assigned to the background label.

This sub-step also involves the use of a morphological filter to check for consistency with the surrounding neighbors of each superpixel. In principal, one object should occupy several superpixels, assuming it is much bigger than any individual superpixel. Therefore, the underlying assumption of morphological filtering is that each classified superpixel should have at least two superpixels in the eight-superpixel-neighborhood (eight closest superpixels from the center superpixel) classified as the same class. Otherwise, the classified superpixel is re-assigned to the most prominent class label in the eight-superpixel-neighborhood. A potential drawback of this approach is that the final classification results may tend towards dominant, contiguous classes, thus reducing the representation of rare classes. An alternate way of approaching this neighborhood consistency can be using the proposed method by Fulkerson et al. [32]. They used conditional random field to refine the classification

4. PROPOSED METHOD

result to increase the classification accuracy. Spatial regularization and classification confidence are used in this method to generate a more accurate thematic mapping of the mosaic. This particular method can generate a similar accuracy gain as the one with morphological filtering implemented in our method.

4.3 Implementation

The implementation of the proposed framework follows the flowchart illustrated in Figure 4.19. A unique feature of our approach is that it is based on a configurable implementation scheme. Under this scheme, different options are available for each of the main processing steps. The configurable scheme allows tuning the processing pipeline according to the characteristics of different datasets.

As illustrated in Figure 4.19, the raw survey images are initially preprocessed as appropriate. Using these preprocessed images, training images are created. Features are extracted and modified from the training images. Using these modified features, the learning of the classifier is performed following the methods described in the earlier sections of this chapter.

There are several parameters in each option of the proposed classification framework that need to be tuned beforehand. To select these parameters, comprehensive testing was done on underwater optical image datasets such as *EILAT*, *RSMAS*, *MLC 2008* and *EILAT 2* (Appendix A). The recommendations to make this selection of an appropriate configuration for the proposed framework is given in Chapter 5. Once the parameters of all the sub-steps were tuned, they were reused for other underwater optical image datasets.

4.4 Summary

This chapter presented the proposed framework for underwater image classification with implementation details. Most of the methods used in the different steps of the framework are inspired from the state of the art techniques in the computer vision field and adopted to the underwater optical domain.

4. Proposed method

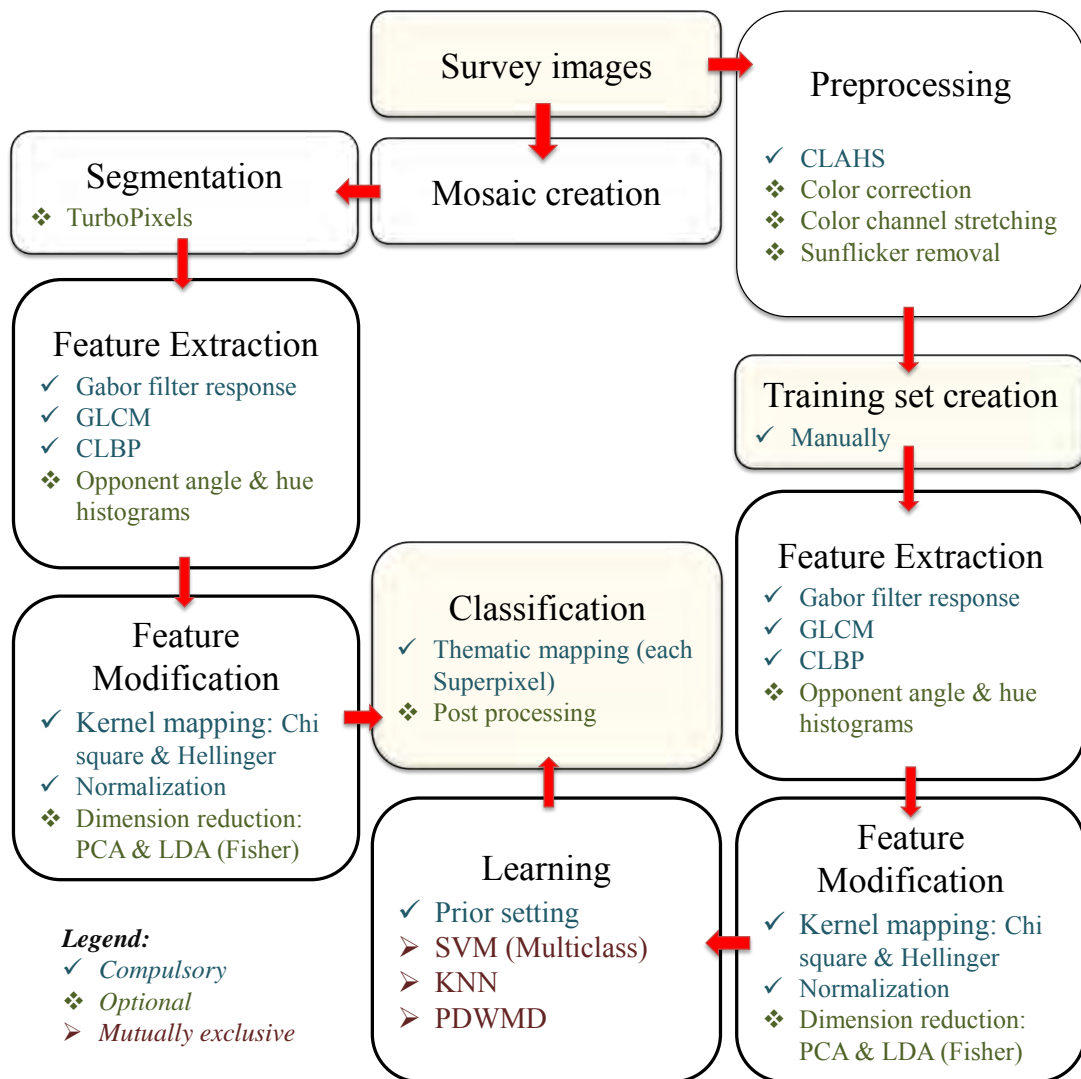


Figure 4.19: Flowchart of the implementation of the proposed method.

4. PROPOSED METHOD

To work with any raw survey image, an initial preprocessing is performed to improve image comparability. It also improves the visibility of the image features by counteracting some of the effects of the medium, such as light scattering, color shifts, blurring and many more. The image enhancement step contains one sub-step that is always used (CLAHS) and three optional sub-steps (color correction, color stretching and sunflicker removal) to be used as appropriate for the dataset. The preprocessing is applied to both the training image set (created manually by experts), and the mosaic to be classified.

The gabor filter response [81], the grey level co-occurrence matrix (GLCM) [8; 43; 98], and the completed local binary pattern (CLBP) [40] are used as texture descriptors, and the opponent angle, hue color histograms (optional) [107], are used as color descriptors. Kernel mapping and normalization are used as a compulsory sub-step, where dimension reduction is used as an optional sub-step. In the learning step, any one of the algorithms, the support vector machine (SVM) [12], the k nearest neighbor (KNN) [14] or the probability density weighted mean distance (PDWMD) [100] is chosen. Finally, the knowledge base learned is used to generate segmented and classified thematic maps of mosaic images of the seabed.

Chapter 5

Results and Analysis

This chapter presents the evaluation criteria, experimental results and analysis of our method. The experimental setup described in the following section is the quantitative assessment of the proposed method with respect to other related works. Insights on the selection of appropriate configurations are extracted and presented, after analyzing the method's performance on adopted experimentation. Additionally, this chapter illustrates the creation of thematic maps created using the framework proposed.

5.1 Methodology for evaluation

This section presents the configuration of several experiments which were carried out to assess the primary aspects of the method.

5.1.1 Classification performance evaluation

Two widely used indicators (the error matrix and the precision-recall curve) are used to evaluate the classification quality of each compared method in this thesis.

The error matrix quantifies the accuracy of each classified category as a percentage based on the total number of validation points in each category [11]. The

5. RESULTS AND ANALYSIS

overall accuracy (OA) of the classified dataset is defined as the sum of the number of correctly classified examples divided by the total number of tested examples.

The precision-recall curve is computed by varying the decision threshold (from high to low) and plotting the values of precision against recall for each threshold value [82]. Precision is defined as the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. The average precision (AP) summarizes the precision-recall curve by measuring the area under the curve. High recall indicates that an algorithm is capable of returning most of the relevant results. High precision indicates that an algorithm returns more relevant results than irrelevant.

For the accuracy evaluation of the thematic mapping, kappa statistics along with overall accuracy are used to assess the performance of our proposed method. These measurement methods were discussed earlier in Chapter 2.

5.1.2 Configuration selection

Our insights on how to decide which configurations to use (based on the properties of any new dataset) are supported by the results presented in section 5.2. The parameters for all the methodologies are tuned using the four underwater optical image datasets (EILAT, RSMAS, MLC 2008 and EILAT 2) as described in Appendix A. These tuned parameters can be used directly for a new underwater image dataset.

Various combinations of sub-steps with appropriate parameters can achieve the highest classification performance out of a particular image dataset. Therefore, the right selection from the various options available in each step is critical. A series of experiments were conducted varying the choice of options within each step, while keeping the options for the other steps constant. In all the experiments, for each possible configuration, we used 10 cross-fold random validation, meaning that 90% of the image patches were used for training and 10% were used for testing over 10 iterations. With cross-fold random validation, we reduced the possibility of having results biased by over-fitting [53].

When our method is compared with other methods, those methods were also tuned to the prescribed parameters in their respective publications ensuring possible best performance of each.

5.1.3 Dataset characteristics

The characteristics of the test datasets are presented in Table 5.1. More details are given in Appendix A. Datasets with less than 4,000 training samples are considered as being small. Datasets with more than 9,000 training samples are considered large. Datasets with more than 4,000 samples but less than 9,000 are considered medium sized.

Table 5.1: A brief summary of the characteristics of the underwater image datasets used for classification and thematic mapping.

| Name | Number of classes | Number of samples | Samples used for training | Sample resolution | Color | Type | Size |
|-------------------------|-------------------|-------------------|---------------------------|-------------------|-------|---------|--------|
| <i>EILAT</i> | 8 | 1,123 | 561 | 64×64 | Yes | Benthic | Small |
| <i>RSMAS</i> | 14 | 766 | 383 | 256×256 | Yes | Benthic | Small |
| <i>EILAT 2</i> | 5 | 303 | 151 | 128×128 | Yes | Benthic | Small |
| <i>MLC 2008</i> | 9 | 18,872 | 9,436 | 312×312 | Yes | Benthic | Large |
| <i>BMAT</i> | 3 | 37,759 | 7,550 | 192×192 | No | Benthic | Medium |
| <i>UIUCtex</i> | 25 | 1,000 | 500 | 640×480 | No | Texture | Small |
| <i>CURET</i> | 61 | 5,612 | 2,806 | 200×200 | No | Texture | Small |
| <i>KTH-TIPS</i> | 9 | 810 | 405 | 200×200 | No | Texture | Small |
| <i>Red Sea mosaic</i> | 6 | 73,600 | 1,426 | 64×64 | Yes | Mosaic | Small |
| <i>North Sea mosaic</i> | 3 | 1,063,000 | 7,550 | 192×192 | Yes | Mosaic | Medium |
| <i>Grounding mosaic</i> | 2 | 13,762 | 4,083 | 180×180 | Yes | Mosaic | Medium |

5.1.4 Thematic mapping accuracy measurement

Our proposed method and four most relevant algorithms (as these algorithms used only optical images for classification like ours and showed reasonable performance over multiple object classes) are applied to two underwater mosaic images to create thematic maps of contiguous sections of the seabed, covering an area much larger than a single image. The two mosaics (the *Red Sea* and *North Sea* mosaics) are presented at the end of this chapter in section 5.3. The *North Sea mosaic* image

5. RESULTS AND ANALYSIS

is mainly used to evaluate the applicability of thematic mapping directly to the generated mosaic rather than applying it to individual images.

Ground truths of the *Red Sea* and *North Sea* mosaics were created manually to examine the effects of morphological filtering. We use these ground truths for a quantitative assessment of the proposed method and the other four algorithms. The resultant thematic maps are compared with corresponding ground truths based on the overall accuracy (OA), average precision (AP), kappa (K) and the average mutual information (AMI) indicators.

5.2 Experimentation with different configurations

This section analyzes the performance of different configurations on four benthic datasets.

5.2.1 Pre-processing sub-steps

Pre-processing sub-steps, such as color correction and sunflicker removal, are implemented whenever possible (i.e. whenever there are color references in the images with the presence of large refracted sunlight pattern). We experimented with possible combinations of sub-steps in this step, keeping the rest of the steps as follows:

- Feature extraction: color histograms, CLBP, Gabor and GLCM,
- Feature modification: normalization & PCA,
- Classification: KNN, class frequency as prior settings.

Among the image pre-processing sub-steps, CLAHS alone produced the best results for the EILAT, RSMAS and EILAT 2 datasets as given in Table 5.2. The combined effect of CLAHS, color correction and color stretching produced the best result for the MLC 2008 dataset. It shows that CLAHS performs consistently well on benthic datasets, therefore can be considered as a mandatory pre-processing sub-step. Other sub-steps in this step, such as color correction, color stretching and sunflicker removal, can be used whenever possible.

5. Results and Analysis

Table 5.2: Percentage of overall accuracy with different image enhancement options as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. The configurations for the other steps are fixed as follows. Feature extraction: completed local binary pattern (CLBP), grey level co-occurrence matrix (GLCM), Gabor filter response, opponent angle and the hue channel histogram; Feature modification: normalization, principal component analysis (PCA); prior: class frequency; classifier: k-nearest neighbor (KNN). NA stand for not applicable.

| Image pre-processing configurations | EILAT | RSMAS | EILAT 2 | MLC 2008 |
|---|-------|-------|---------|----------|
| No pre-processing | 90.7 | 70.1 | 80.1 | 64.0 |
| Color correction | NA | NA | NA | 63.8 |
| CLAHS | 92.9 | 85.8 | 87.4 | 69.3 |
| Color stretching | 67.1 | 58.8 | 62.9 | 70.5 |
| CLAHS + Color correction | NA | NA | NA | 70.9 |
| CLAHS + Color stretching | 91.4 | 82.4 | 81.7 | 72.7 |
| CLAHS + Color correction + Color stretching | NA | NA | NA | 73.2 |

5.2.2 Feature extraction sub-steps

Feature extraction sub-steps are analyzed by varying among possible combinations of hue and opponent angle color histograms, CLBP, Gabor filter response and GLCM. The sub-steps of the other steps are kept constant, and are set as follows: CLAHS for pre-processing, normalization & PCA in the feature modification step, class frequency as prior class probability and KNN as classifier.

- Pre-processing: CLAHS,
- Feature modification: normalization & PCA,
- Classification: KNN, class frequency as prior settings.

Regarding the feature extraction options, the combined feature vector of the color histograms (hue and opponent angle), CLBP, Gabor filter response and GLCM worked most effectively on the EILAT, RSMAS and MLC 2008 datasets (Table 5.3). In general, if color images are available, color histograms can be used together with texture descriptors. However, for the EILAT 2 dataset, the color features are not discriminative enough to aid the classification performance in the combined feature vector. The combination of CLBP, Gabor filter response and GLCM (texture descriptors) works better than any single feature for all four datasets.

5. RESULTS AND ANALYSIS

Table 5.3: Percentage of overall accuracy with different feature extraction methods as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. CLBP, GLCM, Gabor filter response and color histogram (hue + opponent angle) are evaluated in these experiments. In this experiment, fixed configurations on the rest of the steps are as follows. Image pre-processing: CLAHS; feature modification: normalization, PCA; prior: Class frequency; classifier: k-nearest neighbor (KNN).

| Feature extraction configurations | EILAT | RSMAS | EILAT 2 | MLC 2008 |
|--|-------|-------|---------|----------|
| CLBP | 91.3 | 74.7 | 84.8 | 55.1 |
| Gabor | 85.7 | 61.2 | 65.3 | 39.4 |
| GLCM | 70.9 | 62.9 | 58.2 | 46.8 |
| Color histogram (hue + opponent angle) | 64.2 | 81.7 | 53.0 | 41.3 |
| CLBP + Gabor | 90.5 | 75.0 | 87.7 | 54.9 |
| CLBP + GLCM | 92.2 | 75.8 | 86.4 | 57.1 |
| Gabor + GLCM | 87.6 | 72.1 | 77.5 | 46.3 |
| CLBP + GLCM + Gabor filter response | 93.4 | 83.5 | 91.2 | 62.4 |
| CLBP + GLCM + Gabor filter response + color histogram (hue + opponent angle) | 94.7 | 89.6 | 87.3 | 65.7 |

5.2.3 Feature modification and learning

Two separate experiments were performed to determine the best combination of feature modification options. In the first experiment, the possible combinations of the chi-square kernel, Hellinger kernel and L1 normalization were tested while keeping the options in the other steps constant as follows:

- Pre-processing: CLAHS,
- Feature extraction: color histograms, CLBP, Gabor and GLCM,
- Feature modification: PCA,
- Classification: KNN, class frequency as prior settings.

In the second experiment, the combinations of dimension reduction and classifier methods were explored. The use of PCA and the Fisher kernel were varied with each of the three classifiers while keeping the options for the other steps set as follows:

- Pre-processing: CLAHS,
- Feature extraction: color histograms, CLBP, Gabor and GLCM,
- Feature modification: normalization.

5. Results and Analysis

With respect to the available feature modification sub-steps, the best results are obtained by using both the chi-square and Hellinger kernels along with normalization, as illustrated in Table 5.4. For the EILAT and RSMAS datasets, the best classifier and dimension reduction options are the KNN classifier with dimension reduction by PCA and Fisher kernel mapping (Table 5.5). For the EILAT 2 dataset, the SVM classifier without dimension reduction produces the highest overall accuracy. For the MLC 2008 dataset, maximum overall accuracy is achieved with PDWMD classifier and no dimension reduction.

Table 5.4: Percentage of overall accuracy with different normalization and kernel mapping methods, as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. In this experiment, fixed configurations for the rest of the steps are as follows. Image pre-processing: CLAHS; feature extraction: CLBP, GLCM, Gabor and color histograms; PCA; prior: class frequency; classifier: k-nearest neighbor (KNN).

| Feature modification configurations | EILAT | RSMAS | EILAT 2 | MLC 2008 |
|---|-------|-------|---------|----------|
| No feature modification | 87.5 | 84.8 | 87.3 | 61.9 |
| L1 normalization | 91.9 | 87.6 | 87.4 | 64.0 |
| Chi-square kernel mapping | 85.7 | 87.6 | 88.1 | 63.3 |
| Hellinger kernel mapping | 85.8 | 87.9 | 88.5 | 62.2 |
| Kernel mapping (chi-square, Hellinger) | 89.2 | 88.1 | 89.3 | 65.7 |
| L1 normalization + kernel mapping (chi-square, Hellinger) | 93.4 | 89.7 | 91.1 | 66.5 |

5.2.4 Selected configurations and recommendations for future datasets

Using the experimental results above (Tables 5.2, 5.3, 5.4 & 5.5) we selected the configurations shown in Table 5.6. The experimental results above and tables 5.1 and 5.6 provide the following insights concerning future image classifications:

1. If the dataset contains low contrast or blurred images, CLAHS works very well for contrast enhancement. Therefore, CLAHS can be used for any future challenging dataset. If color markers are available in raw images, color correction can be performed to enhance the color constancy. Color stretching and sunflicker removal sub-steps are to be used on datasets whenever applicable.
2. Segmentation is only applicable for large scale or mosaic images whenever thematic mapping is performed.

5. RESULTS AND ANALYSIS

Table 5.5: Percentage of overall accuracy with different dimension reduction and classification methods (SVM with radial basis kernels having one to all schemes, KNN or PDWMD) as evaluated with the EILAT, RSMAS, EILAT 2 and MLC 2008 datasets. In the table, PCA, Fisher kernel (F), a combination of principal component analysis and Fisher kernel (P+F) and no dimension reduction (ND) are applied. In this experiment, fixed configurations on the rest of the steps are as follows. Image pre-processing: CLAHS; feature extraction: CLBP, GLCM, Gabor, color histograms; feature modification: normalization; prior: class frequency.

| EILAT | | | | |
|-----------------|-------------|------|------|-------------|
| | ND | PCA | F | P+F |
| SVM | 94.3 | 91.9 | 86.7 | 90.2 |
| KNN | 91.7 | 93.4 | 85.4 | 94.9 |
| PDWMD | 91.2 | 90.5 | 84.9 | 87.1 |
| RSMAS | | | | |
| | ND | PCA | F | P+F |
| SVM | 92.1 | 89.0 | 81.8 | 88.2 |
| KNN | 91.4 | 92.8 | 87.4 | 93.5 |
| PDWMD | 87.9 | 86.5 | 85.5 | 84.3 |
| EILAT 2 | | | | |
| | ND | PCA | F | P+F |
| SVM | 93.1 | 91.5 | 88.9 | 90.7 |
| KNN | 88.0 | 92.2 | 85.1 | 92.7 |
| PDWMD | 89.2 | 89.4 | 80.7 | 85.1 |
| MLC 2008 | | | | |
| | ND | PCA | F | P+F |
| SVM | 75.2 | 71.7 | 61.2 | 70.1 |
| KNN | 69.5 | 73.9 | 64.5 | 74.9 |
| PDWMD | 79.8 | 73.3 | 66.6 | 71.3 |

5. Results and Analysis

Table 5.6: Selected configuration of our proposed method for all the datasets used in experimentation. *EILAT*, *RSMAS*, *EILAT 2*, *MLC 2008*, *BMAT*, *CURET*, *KTH*, *UIUC*, *the Red Sea mosaic*, *the North Sea mosaic* and the *Grounding mosaic* are represented with *E*, *R*, *E2*, *M*, *B*, *C*, *K*, *U*, *RM*, *NM* and *GM* respectively. F and P represents class frequency and equal probability respectively.

| Step | Sub-step | <i>E</i> | <i>R</i> | <i>E2</i> | <i>M</i> | <i>B</i> | <i>C</i> | <i>K</i> | <i>U</i> | <i>RM</i> | <i>NM</i> | <i>GM</i> |
|----------------------|--------------------|----------|----------|-----------|----------|----------|----------|----------|----------|-----------|-----------|-----------|
| Pre-processing | Color correction | N | N | N | Y | N | N | N | N | N | N | N |
| | CLAHS | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | Color stretching | N | N | N | Y | N | N | N | N | N | N | N |
| | Sunflicker removal | N | N | N | N | N | N | N | N | N | N | Y |
| Segmentation | Turbopixels | N | N | N | N | N | N | N | N | Y | Y | Y |
| Feature Extraction | Gabor response | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | GLCM | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | CLBP | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | Color histogram | Y | Y | Y | Y | N | N | N | N | Y | N | Y |
| Feature modification | Chi-square kernel | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | Hellinger kernel | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | Normalization | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| | PCA and LDA | Y | Y | Y | N | N | Y | Y | Y | Y | N | N |
| Learning | Prior settings | F | F | F | F | P | P | P | P | F | F | P |
| | SVM | N | N | N | Y | N | N | N | N | N | N | N |
| | KNN | Y | Y | Y | N | N | Y | Y | Y | Y | N | N |
| | PDWMD | N | N | N | N | Y | N | N | N | N | Y | Y |
| Classification | Thematic mapping | N | N | N | N | N | N | N | N | Y | Y | Y |
| | Post processing | N | N | N | N | N | N | N | N | Y | Y | Y |

5. RESULTS AND ANALYSIS

3. The joint use of GLCM, Gabor filter response and CLBP works consistently well as a texture descriptor. Opponent angle and hue color channel histograms can be added to the texture descriptor (assigning equal weights to both color and texture descriptors) if the test images have distinctive and reliable color information.
4. In all cases of image patch classification, sparsely populated bins within histograms possess higher distinctive information than high frequency bins [98]. This statement is based on the assumption that high frequency bins often represent the background of an object and contain less distinctive information. The chi-square and Hellinger kernels can be used to modify bin counts and boost the population of low frequency bins. Normalization of the feature vector is necessary in all cases before applying the classifier for training and testing. Combined use of PCA and Fisher kernel mapping work very effectively to reduce the feature dimension if the dataset size is small (training samples of less than 4,000). However, for medium (training samples of more than 4,000 but less than 9,000) and large datasets (training samples of more than 9,000) almost all the dimensions become discriminative, and only a few dimensions are reduced with PCA. Therefore, for medium and large datasets, this dimension reduction sub-step can be avoided.
5. Class frequency works well as a prior in all cases. However, for texture datasets, equal probability can be used as a prior as each class has the same number of samples in its training set. For *BMAT* and the *Grounding mosaic* datasets, equal probability might be used as well so that the classes that possess less class frequency in their training sets are given the same opportunity as other classes. For small datasets, the KNN classifier provides the best performance. However, the effectiveness of this method diminishes as the datasets get larger, as discussed in Section 4.2.5. For medium sized underwater image datasets, PDWMD works best, both in terms of time and accuracy. For large datasets, SVM can be considered as the appropriate classifier and should be used.
6. Morphological filtering can increase the accuracy of thematic mappings. These post processing sub-steps are used for large images or mosaics where neighborhood information of each patch is available.

5.3 Evaluation of the proposed method

The proposed image classification method is initially evaluated using the *MLC 2008* dataset by constructing a precision-recall curve and an error matrix.

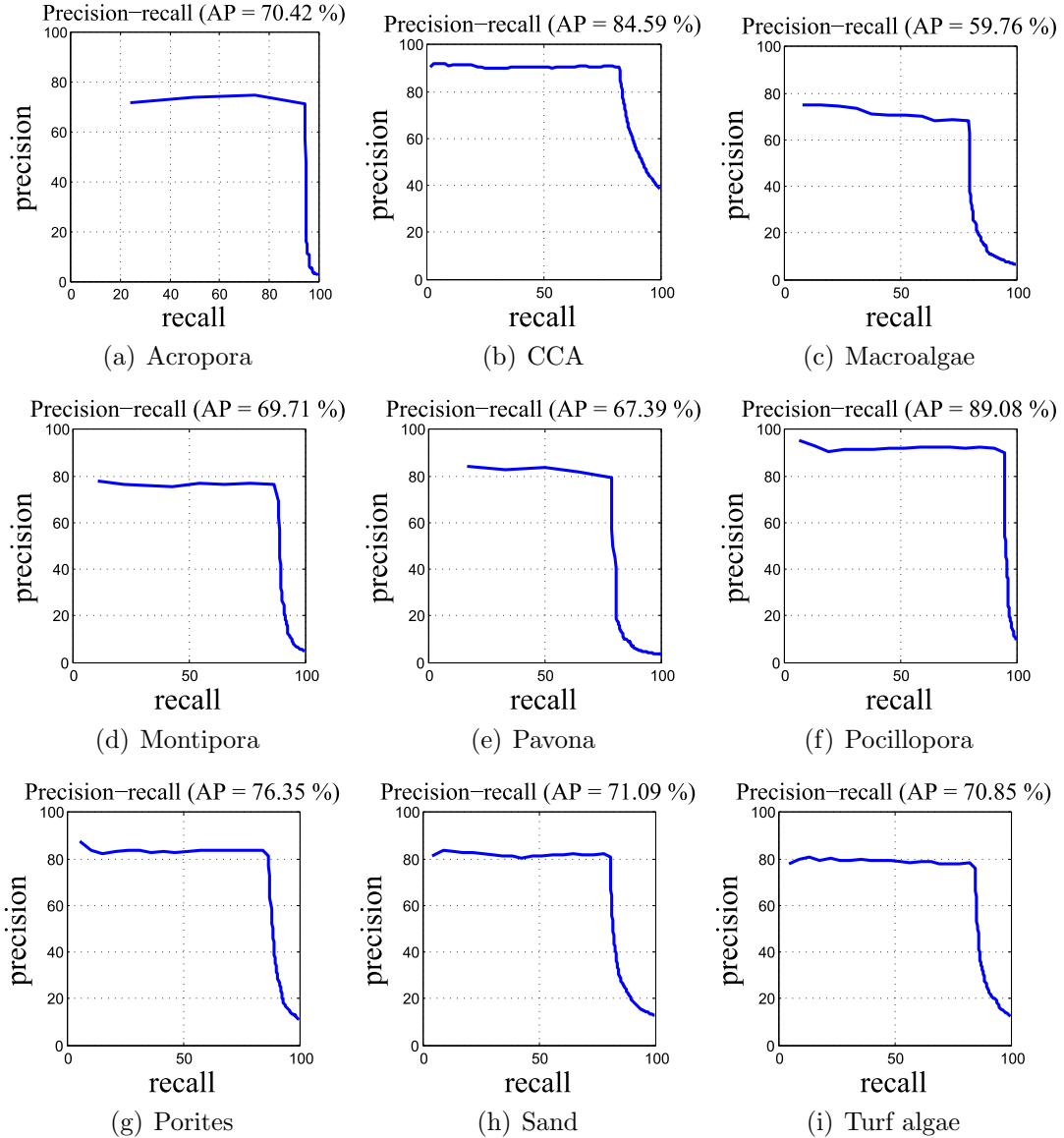


Figure 5.1: Precision-recall curves for individual classes of the *MLC 2008* dataset using our method. Average precision over all the classes for this dataset was 74.8%. The highest precision was observed for Pocillopora, and the lowest value was for the macroalgae class. Our method resulted in 85.5% overall accuracy. In the *MLC 2008* dataset, the highest number of examples was from the CCA class, which also had frequent overlaps with other classes.

Table 5.7 shows the error matrix of our method. The highest number of examples

5. RESULTS AND ANALYSIS

in this dataset is from the crustose coralline algae (CCA) class, which also has frequent common texture features as with other classes. The error matrix reflects this fact, as other classes are highly confused with CCA. Figure 5.1 and Table 5.7 illustrate the precision-recall curves and the error matrix of our proposed method when applied to the *MLC 2008* dataset. Our proposed method results in 85.3% overall accuracy and 75.3% average precision. The highest precision value is observed for Pocillopora, and the lowest for the macroalgae class as shown in Figure 5.1.

Table 5.7: The error matrix of our proposed classification method tested on the *MLC 2008* dataset. The classes in both row and columns corresponds to A = Acropora, CCA = Crustose Coralline Algae, MA = Macroalgae, MO = Montipora, PA = Pavona, PP = Pocillopora, P = Porites, S = Sand and T = Turf. Within the main 9×9 cell portion of the table, the given number corresponds to the raw count of the number of validation image patches that fell into a particular target/output combination.

| | Target Class | | | | | | | | | |
|--|--------------|------------|-------------|------------|------------|------------|------------|------------|------------|------------|
| | | A | C | MA | MO | PA | PP | P | S | T |
| Output Class | A | 146 | 1 | 1 | 0 | 0 | 2 | 2 | 0 | 1 |
| | C | 9 | 2408 | 139 | 51 | 29 | 17 | 38 | 116 | 55 |
| | MA | 4 | 59 | 372 | 0 | 2 | 10 | 5 | 11 | 4 |
| | MO | 0 | 19 | 3 | 336 | 2 | 6 | 2 | 9 | 5 |
| | PA | 0 | 31 | 13 | 4 | 202 | 0 | 0 | 4 | 2 |
| | PP | 14 | 0 | 6 | 3 | 0 | 691 | 11 | 2 | 8 |
| | P | 9 | 32 | 7 | 5 | 3 | 11 | 727 | 11 | 40 |
| | S | 0 | 67 | 3 | 6 | 4 | 1 | 10 | 650 | 60 |
| | T | 6 | 18 | 16 | 11 | 6 | 6 | 24 | 45 | 817 |
| Proposed method: overall accuracy 85.3% | | | | | | | | | | |

5.4 Comparison with other methods

Tables 5.8, 5.9, 5.10 & 5.11 show error matrices for all four compared classification methods on the *MLC 2008* dataset. The proposed framework achieves the highest overall accuracy (OA) and highest average precision for six datasets out of a total of seven (Tables 5.12 and 5.13). For the *EILAT* dataset, all methods work reasonably well, though our method achieves the highest overall accuracy (96.9%). The second highest OA is achieved with the Marcos classification (87.9%) [65]. The results of the *RSMAS* dataset are similar to those of the *EILAT* dataset, except the second most accurate method is the Beijbom classification, with an overall accuracy of 85.4%. For

5. Results and Analysis

the *MLC 2008* dataset, our method results in 85.5% overall classification accuracy (Table 5.7).

Table 5.8: The error matrix of the Marcos method [65] tested on the *MLC 2008* dataset.

| | Target Class | | | | | | | | | |
|--------------|--|-----------|-------------|------------|------------|------------|------------|------------|------------|------------|
| | | A | C | MA | MO | PA | PP | P | S | T |
| Output Class | A | 68 | 28 | 23 | 12 | 0 | 32 | 17 | 1 | 7 |
| | C | 1 | 2226 | 94 | 30 | 24 | 30 | 35 | 84 | 111 |
| | MA | 4 | 251 | 228 | 2 | 8 | 29 | 16 | 0 | 22 |
| | MO | 0 | 100 | 7 | 169 | 0 | 21 | 21 | 14 | 84 |
| | PA | 0 | 33 | 0 | 0 | 197 | 0 | 8 | 4 | 6 |
| | PP | 9 | 69 | 36 | 17 | 0 | 554 | 51 | 0 | 8 |
| | P | 0 | 112 | 28 | 12 | 0 | 60 | 531 | 29 | 67 |
| | S | 0 | 198 | 3 | 10 | 2 | 5 | 25 | 600 | 105 |
| | T | 2 | 156 | 32 | 34 | 0 | 8 | 38 | 94 | 628 |
| | Marcos method: overall accuracy 68.7% | | | | | | | | | |

Table 5.9: The error matrix of the Stokes & Deane method [100] tested on the *MLC 2008* dataset.

| | Target Class | | | | | | | | | |
|--------------|--|------------|-------------|------------|------------|------------|------------|------------|------------|------------|
| | | A | C | MA | MO | PA | PP | P | S | T |
| Output Class | A | 112 | 17 | 12 | 1 | 0 | 21 | 12 | 0 | 13 |
| | C | 3 | 2366 | 62 | 22 | 25 | 19 | 28 | 67 | 43 |
| | MA | 2 | 166 | 318 | 2 | 3 | 28 | 5 | 3 | 33 |
| | MO | 4 | 115 | 7 | 247 | 2 | 1 | 10 | 10 | 20 |
| | PA | 0 | 38 | 8 | 3 | 117 | 0 | 7 | 6 | 9 |
| | PP | 16 | 62 | 21 | 0 | 1 | 625 | 8 | 1 | 10 |
| | P | 9 | 100 | 8 | 8 | 2 | 4 | 629 | 8 | 71 |
| | S | 0 | 178 | 4 | 5 | 2 | 3 | 18 | 686 | 52 |
| | T | 2 | 94 | 12 | 9 | 1 | 4 | 53 | 46 | 771 |
| | Stokes & Deane method: overall accuracy 78.3% | | | | | | | | | |

The proposed method achieves almost similar classification accuracy when tested with traditional and texture-only image datasets (Table 5.12). Most of the other methods however, performed significantly worse against texture-only datasets. The failure of the Marcos [65], Stokes & Deane [100], Pizarro [80] and Beijbom [2] methods on standard texture datasets indicates that these methods rely heavily on only color information that may limit the robustness of classification algorithms. Color may be inconsistent, distorted, attenuated or absent in some underwater optical

5. RESULTS AND ANALYSIS

Table 5.10: The error matrix of the Pizarro method [80] tested on the *MLC 2008* dataset.

| | Target Class | | | | | | | | | |
|---|--------------|-----------|-------------|------------|------------|------------|------------|------------|------------|------------|
| | | A | C | MA | MO | PA | PP | P | S | T |
| Output Class | A | 74 | 37 | 9 | 5 | 6 | 22 | 17 | 2 | 16 |
| | C | 38 | 1888 | 106 | 55 | 31 | 94 | 84 | 178 | 161 |
| | MA | 26 | 140 | 301 | 3 | 10 | 26 | 13 | 16 | 25 |
| | MO | 4 | 81 | 6 | 175 | 13 | 22 | 43 | 35 | 37 |
| | PA | 7 | 37 | 6 | 20 | 147 | 3 | 17 | 4 | 7 |
| | PP | 102 | 136 | 34 | 14 | 10 | 363 | 29 | 13 | 43 |
| | P | 13 | 119 | 22 | 35 | 27 | 27 | 457 | 41 | 98 |
| | S | 6 | 248 | 17 | 28 | 9 | 23 | 52 | 479 | 86 |
| | T | 10 | 185 | 28 | 21 | 7 | 38 | 87 | 92 | 524 |
| Pizarro method: overall accuracy 58.2% | | | | | | | | | | |

Table 5.11: The error matrix of the Beijbom method [2] tested on the *MLC 2008* dataset.

| | Target Class | | | | | | | | | |
|---|--------------|------------|-------------|------------|------------|------------|------------|------------|------------|------------|
| | | A | C | MA | MO | PA | PP | P | S | T |
| Output Class | A | 148 | 24 | 9 | 2 | 2 | 27 | 16 | 0 | 8 |
| | C | 6 | 2790 | 68 | 35 | 25 | 28 | 66 | 129 | 148 |
| | MA | 2 | 194 | 421 | 15 | 10 | 21 | 10 | 1 | 28 |
| | MO | 1 | 128 | 15 | 309 | 4 | 4 | 20 | 15 | 25 |
| | PA | 1 | 50 | 4 | 1 | 216 | 9 | 21 | 2 | 7 |
| | PP | 5 | 52 | 17 | 5 | 6 | 815 | 16 | 1 | 13 |
| | P | 8 | 150 | 17 | 13 | 10 | 20 | 740 | 31 | 69 |
| | S | 0 | 255 | 1 | 4 | 2 | 4 | 36 | 823 | 61 |
| | T | 3 | 320 | 25 | 13 | 1 | 24 | 69 | 61 | 724 |
| Beijbom method: overall accuracy 73.7% | | | | | | | | | | |

Table 5.12: Comparison based on overall accuracy (OA) (%) for each method/dataset. The highest overall accuracy obtained for each dataset is shown in bold. NA stands for 'Not Applied'.

| | Marcos | Stokes & Deane | Pizarro | Beijbom | Caputo | Zhang | Our |
|-----------------|--------|----------------|---------|---------|--------|-------------|-------------|
| <i>EILAT</i> | 87.9 | 75.2 | 67.3 | 69.1 | NA | NA | 96.9 |
| <i>RSMAS</i> | 69.3 | 82.5 | 73.9 | 85.4 | NA | NA | 96.5 |
| <i>MLC 2008</i> | 68.7 | 78.3 | 58.2 | 73.7 | NA | NA | 85.5 |
| <i>CURET</i> | 20.8 | 49.7 | 38.1 | 86.5 | 98.6 | 98.5 | 99.2 |
| <i>KTH</i> | 25.5 | 88.9 | 48.3 | 36.3 | 95.8 | 96.7 | 98.3 |
| <i>UIUC</i> | 14.6 | 56.9 | 19.9 | 32.2 | 98.2 | 99.0 | 97.3 |
| <i>BMAT</i> | 87.8 | 90.1 | 73.1 | 92.2 | NA | NA | 99.7 |

5. Results and Analysis

Table 5.13: Comparison based Average precision (AP) (%) for each method/dataset. The highest average precision for each dataset is shown in bold. NA stands for 'Not Applied'.

| | Marcos | Stokes & Deane | Pizarro | Beijbom | Caputo | Zhang | Our |
|-----------------|--------|----------------|---------|---------|--------|-------------|-------------|
| EILAT | 85.1 | 73.5 | 58.1 | 64.2 | NA | NA | 97.2 |
| <i>RSMAS</i> | 59.2 | 81.2 | 67.1 | 79.9 | NA | NA | 96.2 |
| <i>MLC 2008</i> | 49.5 | 61.3 | 46.4 | 64.9 | NA | NA | 74.8 |
| <i>CURET</i> | 14.7 | 30.1 | 29.1 | 81.5 | 98.2 | 98.1 | 98.4 |
| <i>KTH</i> | 23.5 | 84.8 | 39.7 | 34.6 | 95.7 | 96.4 | 97.7 |
| <i>UIUC</i> | 14.7 | 43.5 | 18.2 | 30.4 | 97.2 | 98.5 | 96.9 |
| <i>BMAT</i> | 84.6 | 89.0 | 71.7 | 89.5 | NA | NA | 99.7 |

datasets.

The two state of the art texture classification algorithms, Caputo [44] and Zhang [110], perform relatively well on the texture datasets. For the *UIUC* dataset, our method attains 97.3% overall accuracy where Zhang’s method [110] performed slightly better with 99.0% overall accuracy. The *UIUC* dataset has higher resolution than the other datasets (with a patch size of 640×480 pixels) and contains images with a lot of background; therefore, dense descriptors, such as those used in our method, might be influenced by the background, which contains less discriminative information about the object class. Our method is oriented towards small patch sized image datasets, which are more suitable for the intended use of creating thematic maps from mosaics.

For *CURET* and *KTH* datasets, which have a smaller patch size of 200×200 pixels, our proposed method achieves higher overall accuracies than the texture-only method (Zhang [110]). We compared our proposed method with their method in the same experimental conditions and reached 99.2% accuracy for the *CURET* dataset and 98.3% for the *KTH* datasets, where the reported state of the art Caputo’s method [44] obtained 98.6% accuracy for the *CURET* dataset and Zhang [110] obtained 96.7% for the *KTH* dataset.

An important aspect when evaluating a classification approach intended to be used in large datasets, is computational efficiency. The classification methods by Marcos and Stokes & Deane are the two most computationally efficient (Figure 5.2). Our method requires less time compared to the Pizarro [80] and Beijbom [2] method. As for the Pizarro method, the total computation time contains the time

5. RESULTS AND ANALYSIS

required to generate a 500 visual word vocabulary (approximately 650 seconds), which can be performed offline reducing the required overall computation time. All these methods are implemented in Matlab 2010a and tested on Intel core I5-2430M CPU with 2.4 GHz speed and 6 GB RAM.

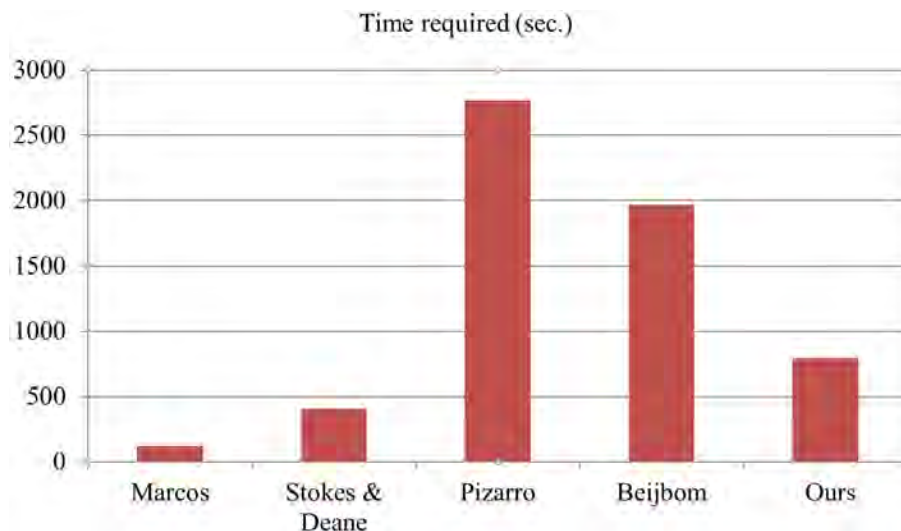


Figure 5.2: The time required to classify the *RSMAS* dataset using four test methods.

The performance consistency is tested as a function of the percentage of training samples used. As the percentage of the training samples for the *MLC 2008* datasets changes from 10% to 60%, accuracy increases gradually for all the methods. This result reflects the fact that a larger number of training samples can produce stronger classifiers (Figure 5.3).

5.5 Classifying mosaic image

In this section, we describe the performance analysis of our proposed method during the creation of a thematic map from mosaic images. We compare the performance against the ground truth annotated manually by experts.

The morphological filtering described in Chapter 4 is used to improve the spatial consistency of the classification using information on the classified neighbors for a given patch, as shown in Figure 5.4. In Figure 5.5, on the left is the input mosaic image and on the right is the segmented and color coded thematic map. Morphological filtering increased the classification accuracy from 82.8% to 83.7%.

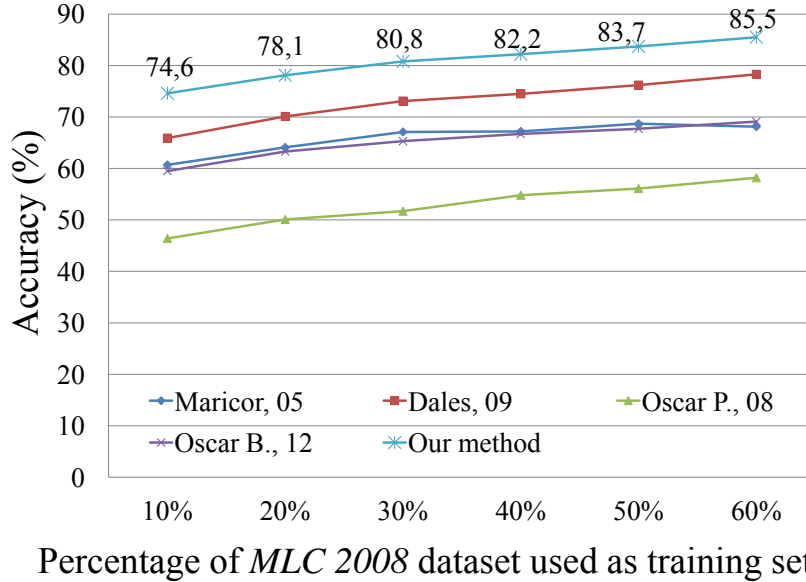


Figure 5.3: The overall accuracy of each method as a function of the number of images in the training data. This test was done on the *MLC 2008* dataset.

It was observed that misclassifications generally tend to be on the borders of the objects as shown in 5.7. This figure shows the location of misclassifications by different methods in red. The classification results of each method when dealing with the *Red Sea* mosaic image are shown in Figure 5.6.

Applying the classification method to segment the full mosaic of the *Red Sea* survey images takes approximately seven hours on an Intel core i5-2430M CPU with 2.4 GHz speed and 6 GB RAM. The image patches are 64×64 pixels with a sliding window with a 16 pixel shift per iteration, resulting in 73,600 patches in total to be classified in the full mosaic.

5.5.1 Thematic mapping directly on mosaic image

We can ascertain that mapping directly onto the mosaic image results in better accuracy compared with creating a mosaic from classified individual images. This is demonstrated by the experimental results provided in Table 5.14. In terms of overall accuracy (OA), average precision (AP) and kappa (K), the classification performed directly on the mosaic yielded at least a 4% increase in accuracy compared to classification of individual images prior to creation of the mosaic. However, cre-

5. RESULTS AND ANALYSIS

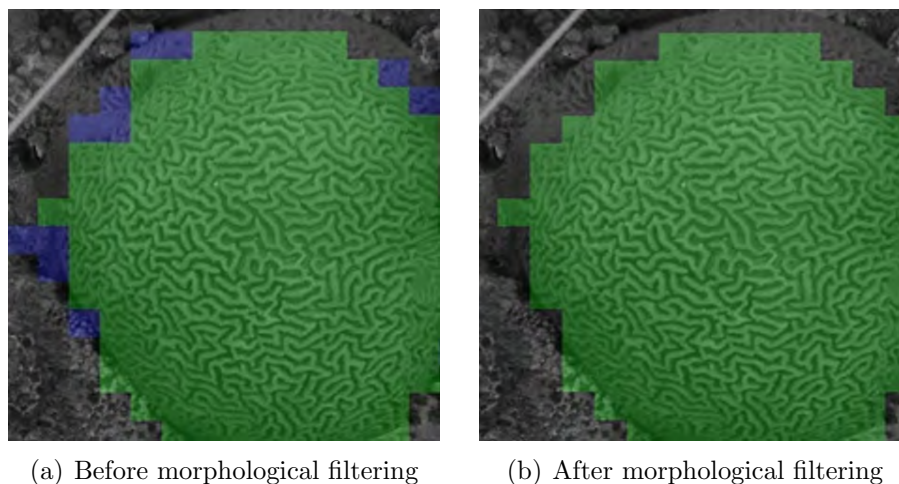


Figure 5.4: Effects of morphological filtering on classification results. (Left) The violet indicates misclassifications, which are removed after morphological filtering (right).

ating the mosaic with classified individual images generates higher average mutual information (AMI). AMI mainly measures the similarity between the ground truth and the classified images. AMI is also dependent on the probability of each class being classified. Therefore, misclassifications of various classes reduces the information gain by various amounts, which is not the case for other accuracy measurement methods, for example overall accuracy (OA). Moreover, the difference between AMI values received in both these schemes are very small.

Figure 5.8 illustrates the thematic mappings of the *North Sea* created using (1) classification directly on the mosaic, (2) classifying individual images and (3) creating the mosaic; and the ground truth. Sand, bacterial mat and shell chaff are labeled in grey, brown and blue, respectively. The ground truth is manually created by experts annotating each of the superpixels individually.

Table 5.14: Comparison based on overall accuracy (OA), average precision (AP), kappa (K) and average mutual information (AMI) between a) Classifying the mosaic image directly and b) Creating a mosaic from the classified images.

| | Mosaic with classified images | Mapping on mosaic image |
|-----|-------------------------------|-------------------------|
| OA | 72.3% | 76.0% |
| AP | 70.8% | 75.1% |
| K | 59.7% | 63.0% |
| AMI | 0.68 | 0.61 |

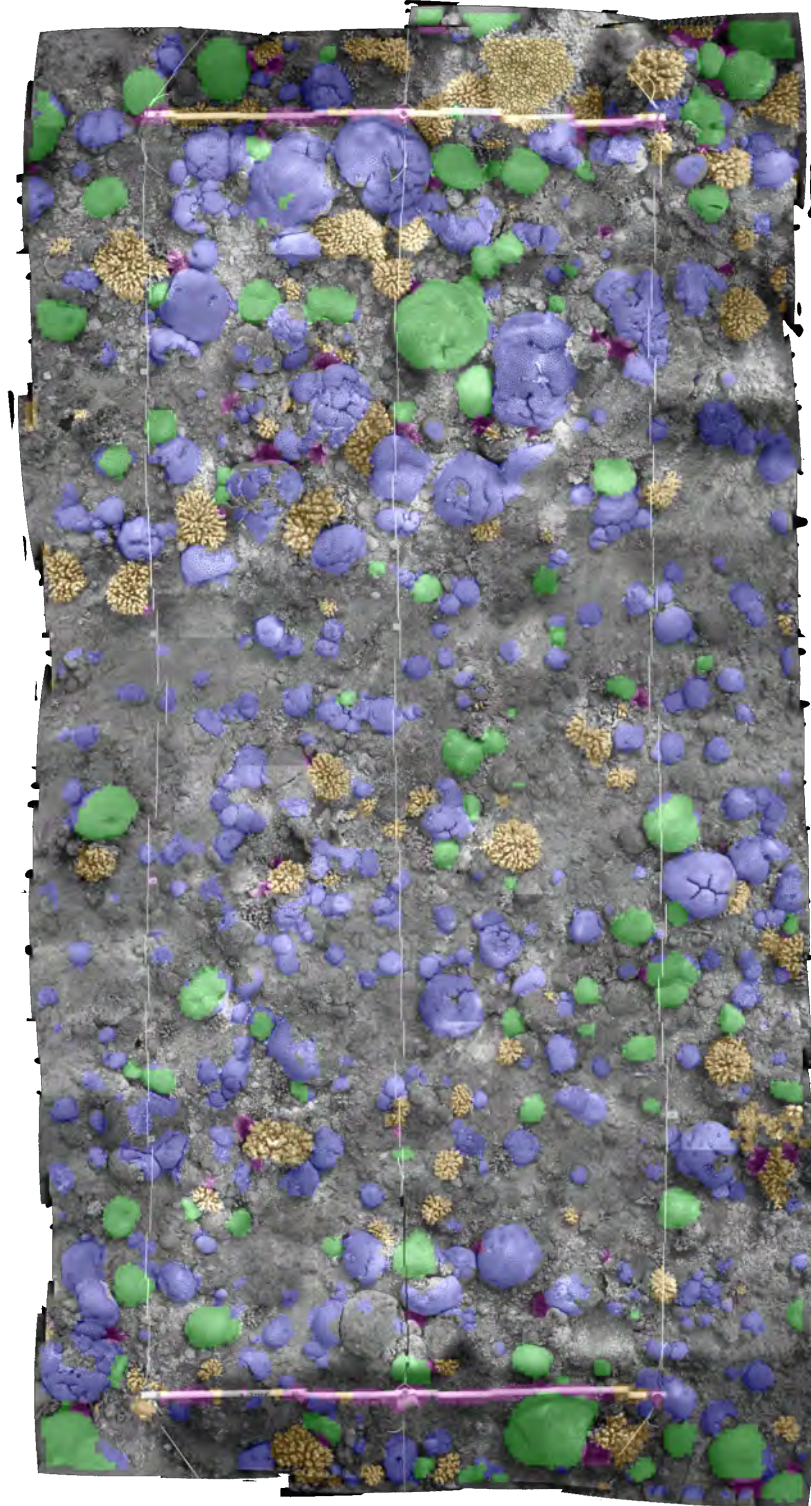
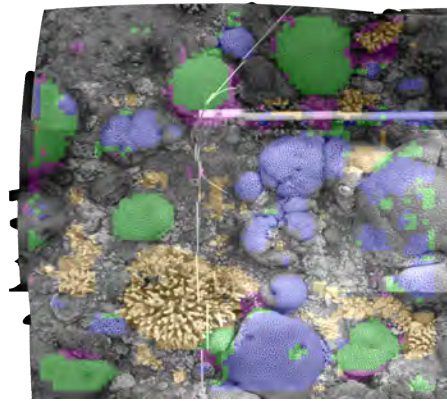
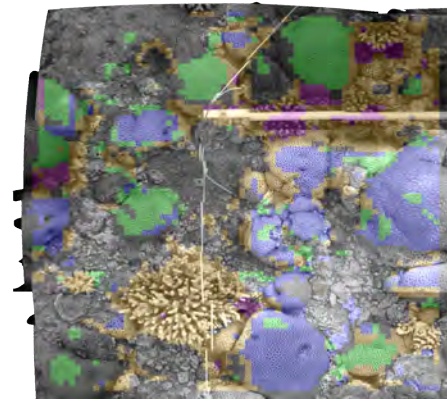


Figure 5.5: Thematic mapping of *Red Sea mosaic* using our proposed method. The segmented images are color coded with the same classification scheme as Figure 5.6.

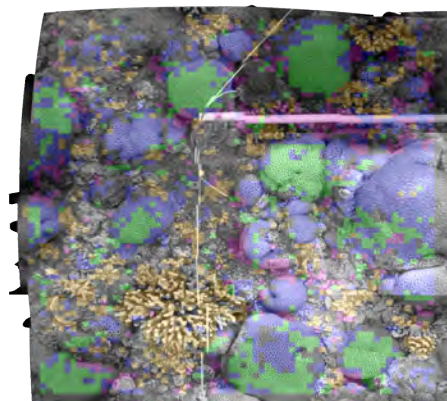
5. RESULTS AND ANALYSIS



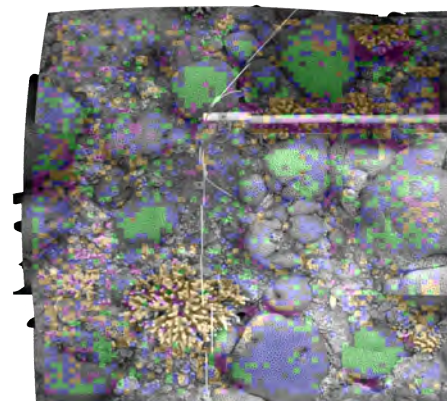
(a) Marcos: accuracy 75.9%



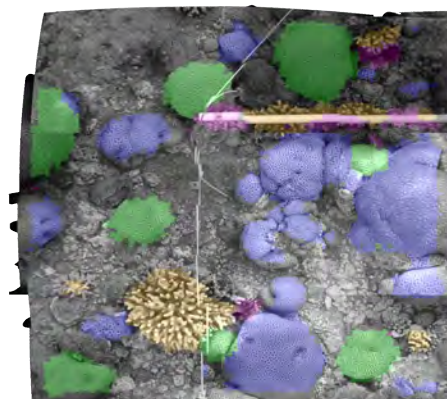
(b) Stokes & Deane: accuracy 67.0%



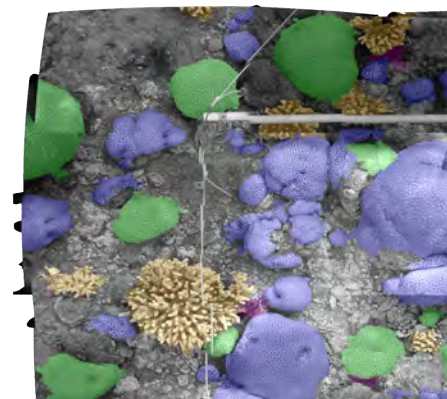
(c) Pizarro: accuracy 68.8%



(d) Beijbom: accuracy 67.4%



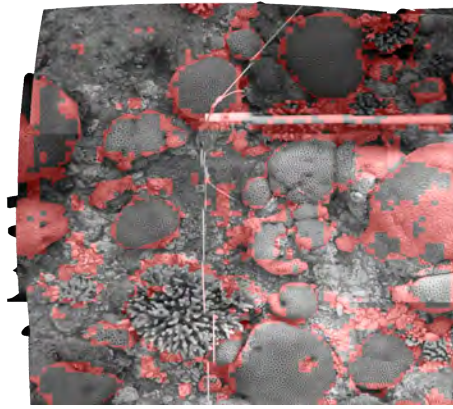
(e) Proposed: accuracy 83.7%



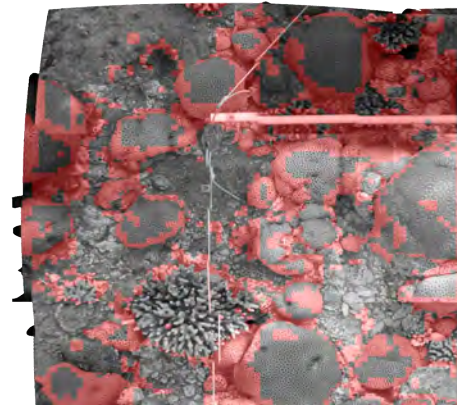
(f) Ground truth

Figure 5.6: The accuracy of the tested classification methods applied to the *Red Sea mosaic*. The segmented images are color coded as: fauid in violet, brain coral in green, branching corals I, II and III in orange, urchin in pink, dead coral and pavement are in grey.

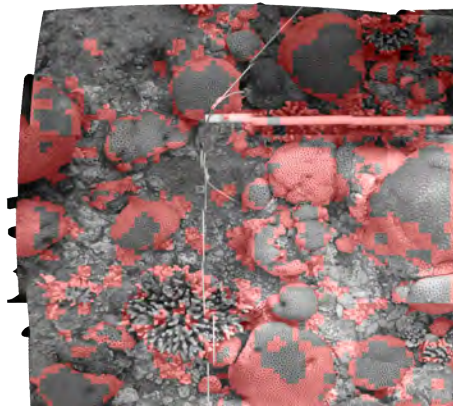
5. Results and Analysis



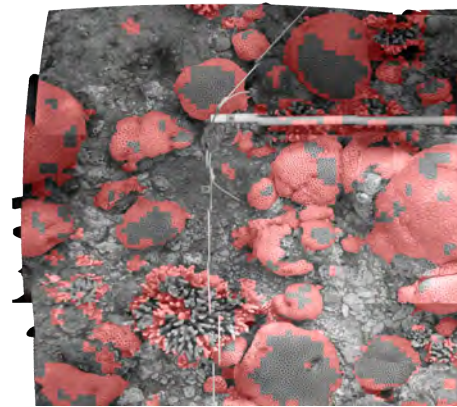
(a) Marcos: accuracy 75.9%



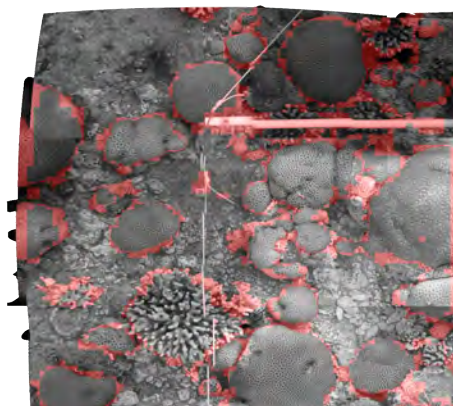
(b) Stokes & Deane: accuracy 67.0%



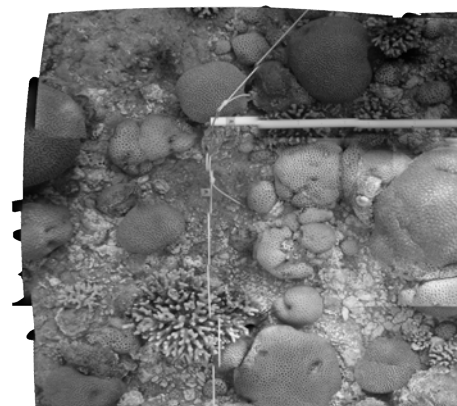
(c) Pizarro: accuracy 68.8%



(d) Beijbom: accuracy 67.4%



(e) Proposed: accuracy 83.7%



(f) Ground truth

Figure 5.7: Location of misclassifications in thematic mapping by different methods shown in red in the *Red Sea mosaic*. The segmented images are color coded as: favid in violet, brain coral in green, branching coral I, II and III in orange, urchin in pink, dead coral and pavement are in grey.

5. RESULTS AND ANALYSIS

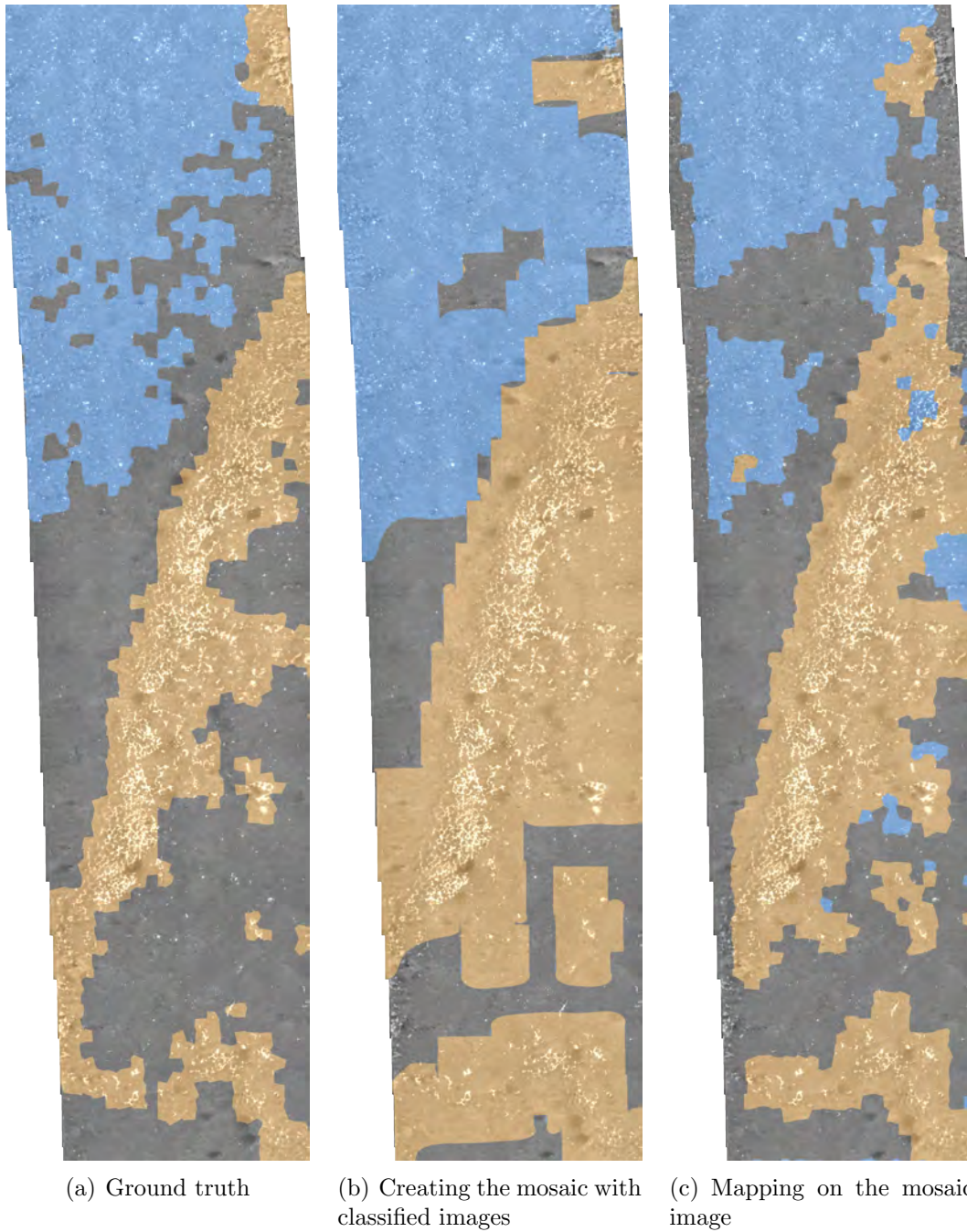


Figure 5.8: Comparison of thematic mapping results for the *North Sea* mosaic. Sand, bacterial mat and shell chaff are labeled in grey, brown and blue respectively. The mosaic covers an area of 232.86 square meters having a width of 7.38 meters and height of 31.57 meters.

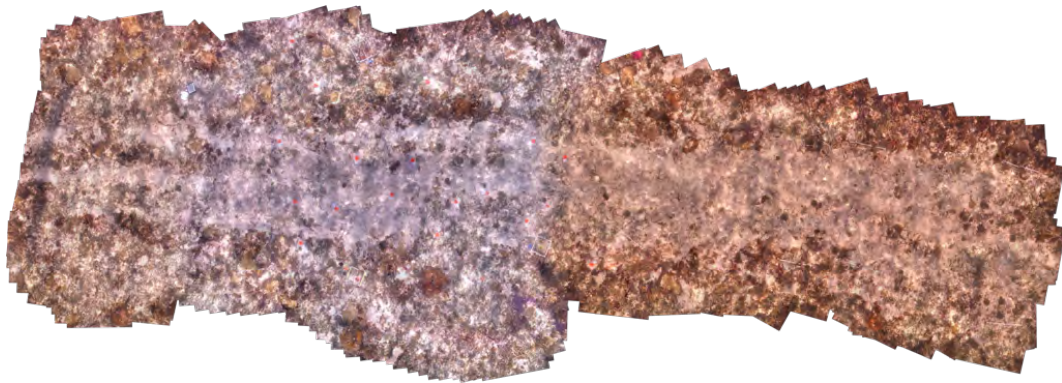
The main benefits of applying thematic mapping directly on the mosaic image are:

- The blending of the registered images [23; 59; 60; 83] tends to reduce artifacts and noise from the mosaic image.
- Some of the objects may only be partially visible in the raw images. However in the mosaic they are fully imaged. Having the objects fully represented helps the classification accuracy, namely when using morphological filtering.
- Classifying directly on the mosaic can save a significant amount of resources in terms of time and memory requirements. In this approach, one object is classified only once, thus avoiding multiple classifications of the same areas across multiple overlapping images.
- Experimental evidence supports the conclusion that better accuracy is achieved when thematic mapping is done directly on the mosaic image.
- Training data is easier to obtain from the experts in the form of an annotated mosaic, than in the form of annotated individual images. Experts get a broader perspective (during annotation) of a patch class in the mosaic facilitating interpretation and analysis.

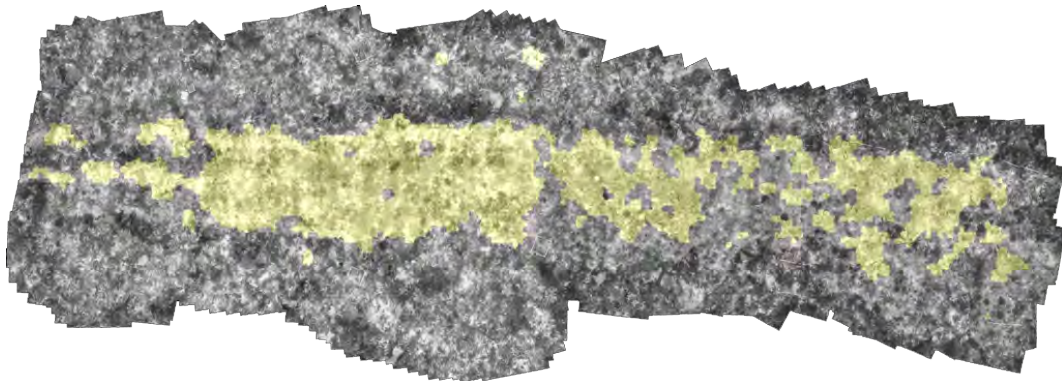
5.5.2 Analysis on sunflicker removed mosaic

The application of the sunflicker removal sub-step in the pre-processing step is illustrated in Figures 5.9 and 5.10. The *Grounding mosaic* dataset contains mosaics with both sunflicker present and sunflicker removed with our proposed method. The thematic map created on the mosaic containing sunflicker presents many misclassifications. However, the thematic map created on the mosaic where the sunflicker has been removed is significantly more accurate and consistent in terms of classification performance with difficult zones. These results illustrate the fact that sunflicker removal can improve the classification performance by a significant amount on a dataset where refracted sunlight is present.

5. RESULTS AND ANALYSIS

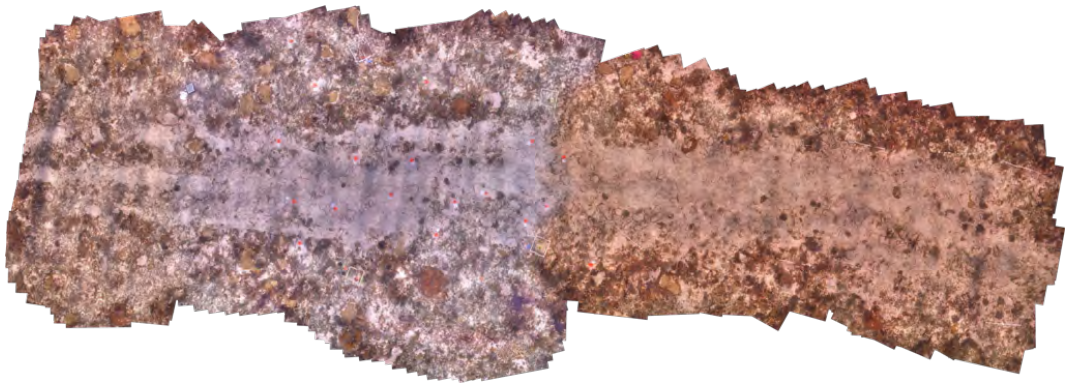


(a) Grounding mosaic without sunflicker removal

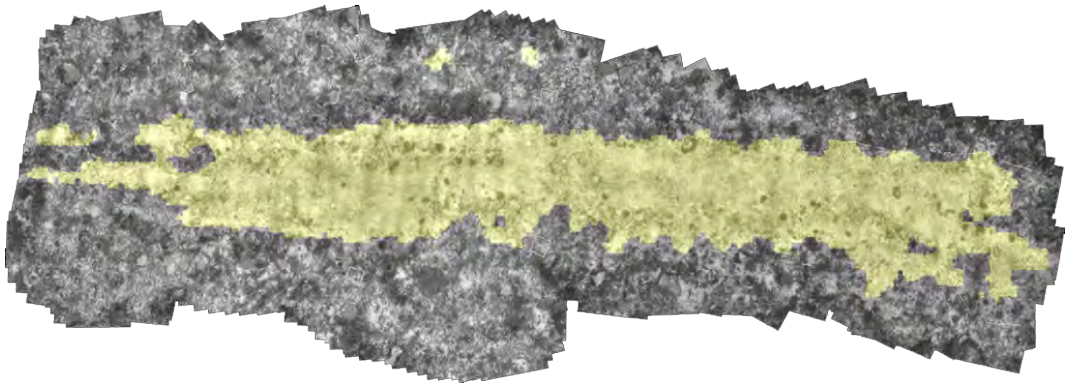


(b) Thematic map created

Figure 5.9: Thematic mapping created on the *Grounding* mosaic without sunflicker removal.



(a) Grounding mosaic with sunflicker removal



(b) Thematic map created

Figure 5.10: Thematic mapping created on *Grounding mosaic* with sunflicker removal.

5. RESULTS AND ANALYSIS

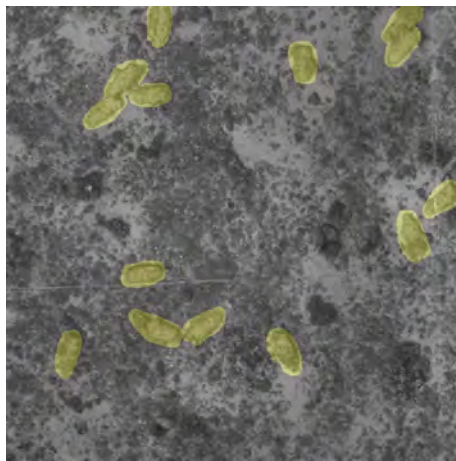
5.5.3 Impact of 3D features

The classification framework was implemented to perform thematic mapping on the Ordnance mosaic and tested on a part of Ordnance dataset used as the validation set (50% of the dataset). The overall classification accuracy and kappa statistics after random 10 crossfold validation are being used for comparing the performance of different methods. The implemented framework have the following components.

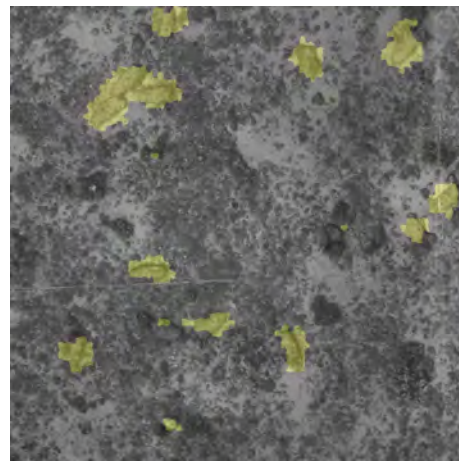
- Pre-processing: CLAHS,
- Segmentation: TurboPixels,
- Feature extraction: Gabor filter response, GLCM, CLBP, Hue and Opponent histograms, 2.5D features,
- Feature modification: Chi-square and Hellinger kernel, PCA, L1 normalization,
- Learning: Probability density weighted mean distance (PDWMD) with equal prior probability,
- Classification: Thematic mapping and morphological filtering as post processing.

The *Ordnance* dataset contains examples of bomb shells which are man-made structured objects appropriate for testing the performance of including 3D features for classification and thematic mapping. Using 2.5D features individually on the *Ordnance* dataset, the classification performance of the validation set and thematic mapping stays moderate. However, using 2D and 2.5D features together provides better results than 2D or 2.5D individually. Figure 5.11 illustrates the thematic mappings created for *Ordnance* dataset using 2D, 2.5D and 2D + 2.5D features.

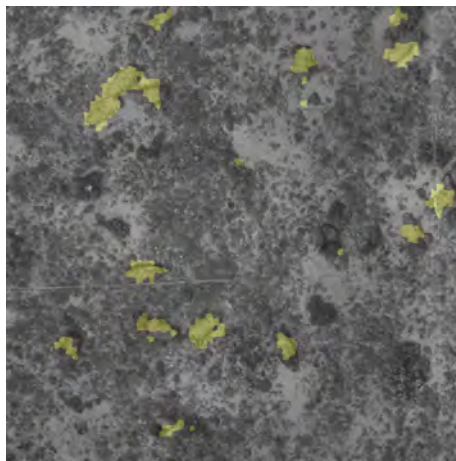
The classification accuracy in this *Ordnance* dataset is very high in general as there are only two classes and most of the samples belong to the background class. Therefore, we can get a better insight about the classification performance using the kappa statistics on the thematic map and average precision on the validation set. Figure 5.12 illustrates the precision-recall curve of background class and bomb shell class separately on the validation set when the combination of 2D and 2.5D features is used.



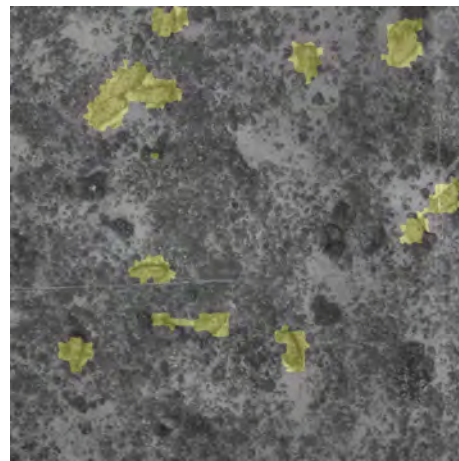
(a) Ground truth



(b) Thematic mapping with 2D features (Kappa 66.51%)



(c) Thematic mapping with 2.5D features (Kappa 55.78%)



(d) Thematic mapping with 2D + 2.5D features (Kappa 70.91%)

Figure 5.11: Thematic mapping with *Ordnance* dataset using 2D, 2.5D and 2.5D + 3D features. Here the yellow color represents the segmented bombshells.

5. RESULTS AND ANALYSIS

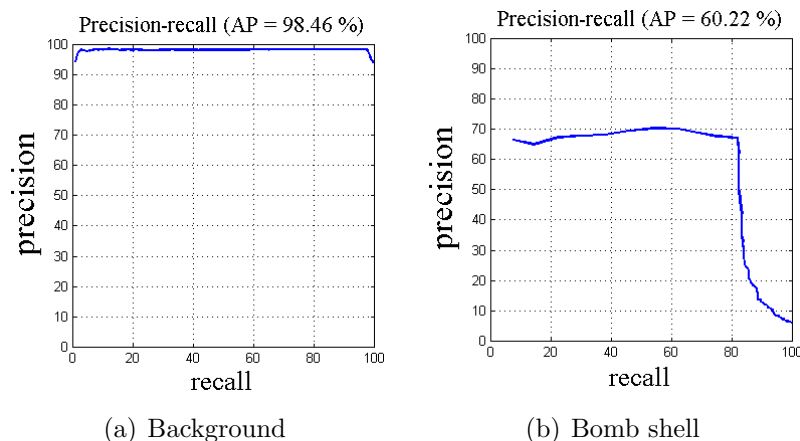


Figure 5.12: Precision-recall curves for both background and bombshells on the validation set of the *Ordnance* dataset when the combination of 2D and 2.5D features is used.

Table 5.15 presents the summary of experimental results for *Ordnance* dataset using 2D, 2.5D and 2D + 2.5D features for classification and thematic mapping. On the thematic mapping, we obtained kappa value of 70.91% using 2D + 2.5D features, whereas using 2D or 2.5D features, we attained only 66.51% and 55.78% kappa respectively as shown in Table 5.15. A similar pattern of results can be observed on the validation set, where the average precision is the highest scoring one with a value of 79.82% using 2D + 2.5D features. This amount of precision is more than 1% higher than the sole use of 2D features. Figure 5.13 illustrates a complete example of classified thematic mapping on the *Ordnance* mosaic.

Table 5.15: Summary and comparison of classification results using 2.5D features with *Ordnance* dataset.

| | | 2D | 2.5D | 2D + 2.5D |
|-------------------------|-----------------------|-----------|-------------|------------------|
| Thematic mapping | Accuracy (%) | 96.06 | 94.59 | 96.51 |
| | Kappa (%) | 66.51 | 55.78 | 70.91 |
| | AMI | 0.146 | 0.107 | 0.165 |
| Validation set | Accuracy (%) | 96.63 | 94.36 | 96.85 |
| | Average precision (%) | 78.51 | 65.24 | 79.82 |

5.5.4 Quantifying change detection

One of the applications of automated thematic mapping can be to detect changes among benthic habitat distribution on the seabed along time. For the *Grounding*

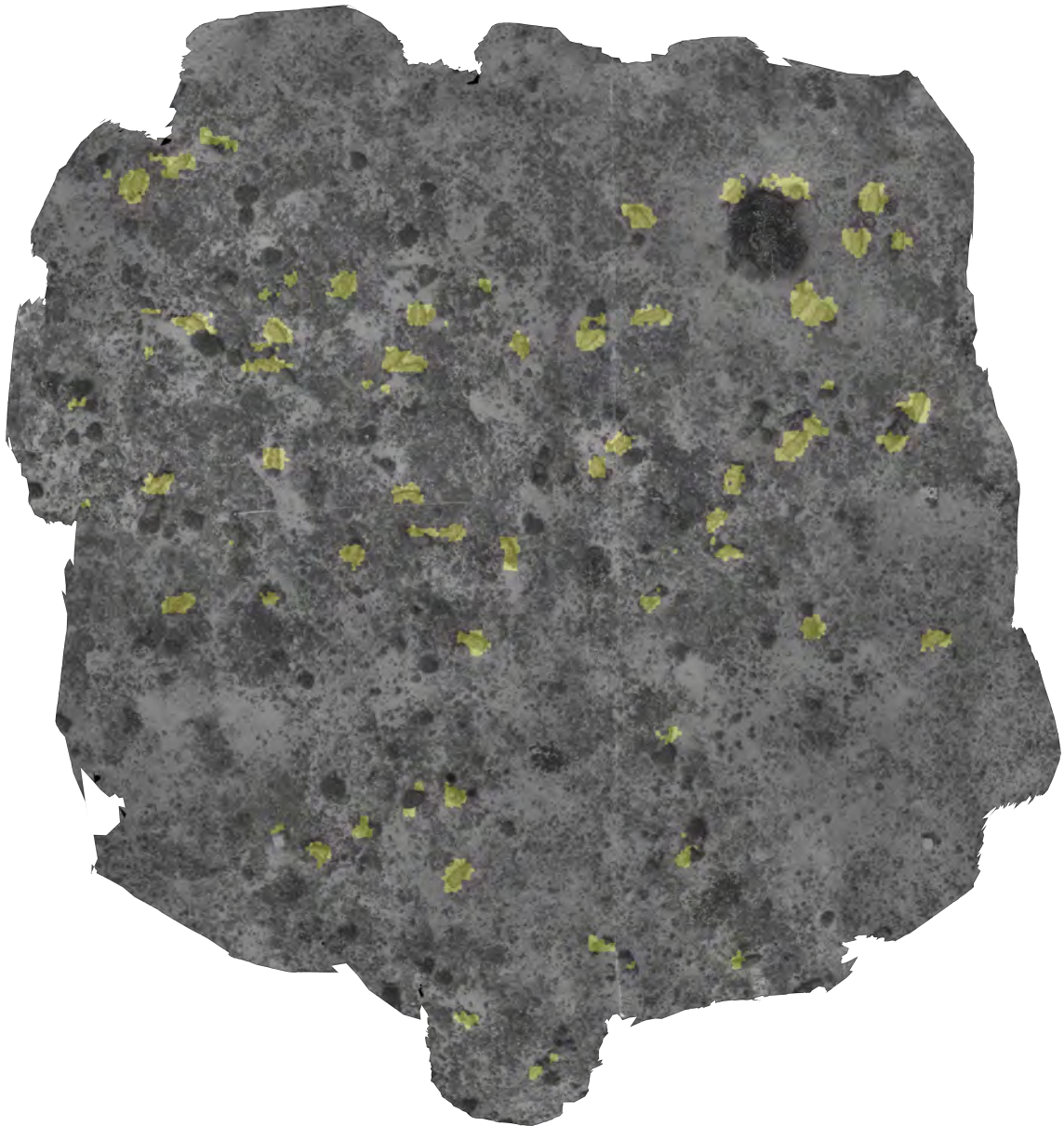


Figure 5.13: Thematic mapping performed on the *Ordnance* mosaic with the use of both 2D and 2.5D features. Here the yellow color represents the segmented bomb shells.

5. RESULTS AND ANALYSIS

mosaic, we have mapped the same area of the ship grounding at the years 2005, 2006, 2011 and 2012. The mosaics from 2011 and 2012 have higher resolution than from the ones of 2005 and 2006. In this experiment, all these four mosaics are segmented and classified to create corresponding thematic mappings. The performance of the thematic mapping method is evaluated with the corresponding ground truths and on the validation sets from the training sets. The created thematic mappings can give indications of how the coral colonies have recovered over the years and how their distribution changed inside the grounding scar areas on the seabed. The summary of the results obtained is shown in the Table 5.16.

Table 5.16: Classification and thematic mapping result for *Grounding mosaic* created over the years of 2005, 2006, 2011 and 2012.

| | | 2005 | 2006 | 2011 | 2012 |
|-------------------------|-----------------------|-------------|-------------|-------------|-------------|
| Thematic mapping | Accuracy (%) | 85.28 | 91.12 | 92.62 | 92.95 |
| | Kappa (%) | 62.49 | 69.30 | 67.86 | 74.93 |
| | AMI | 0.349 | 0.290 | 0.257 | 0.337 |
| Validation set | Accuracy (%) | 91.13 | 89.45 | 96.54 | 93.89 |
| | Average precision (%) | 88.79 | 86.35 | 90.81 | 90.42 |

The thematic mapping on the *Grounding mosaics* shows consistent performance in terms of overall accuracy, kappa and AMI. For the 2005, 2006, 2011 and 2012 mosaic images, the kappa values achieved are respectively 62.49%, 69.30%, 67.86% and 74.93%. Also on the validation sets for corresponding training sets, the average precision values achieved 88.79%, 86.35%, 90.81% and 90.42%. These results as shown in Table 5.16 demonstrate the fact that the automatic classification and segmentation can achieve a reliable mapping of the seafloor depicting the two classes. This automated mapping can be used by the experts to refine into more detailed label. Figures 5.14, 5.15, 5.16 and 5.17 illustrate the thematic mapping results on the *Grounding mosaics* of the years 2005, 2006, 2011 and 2012 respectively.

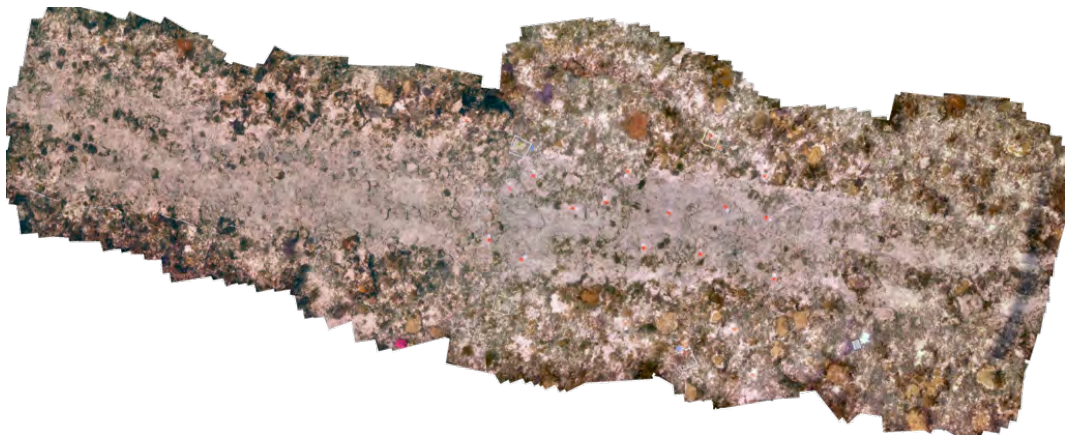
5.6 Closing remarks

This chapter presented the evaluation criteria, experimental results and analysis of the method proposed. The results show that the method, in most cases, can achieve higher classification accuracy than other methods in the state of the art.

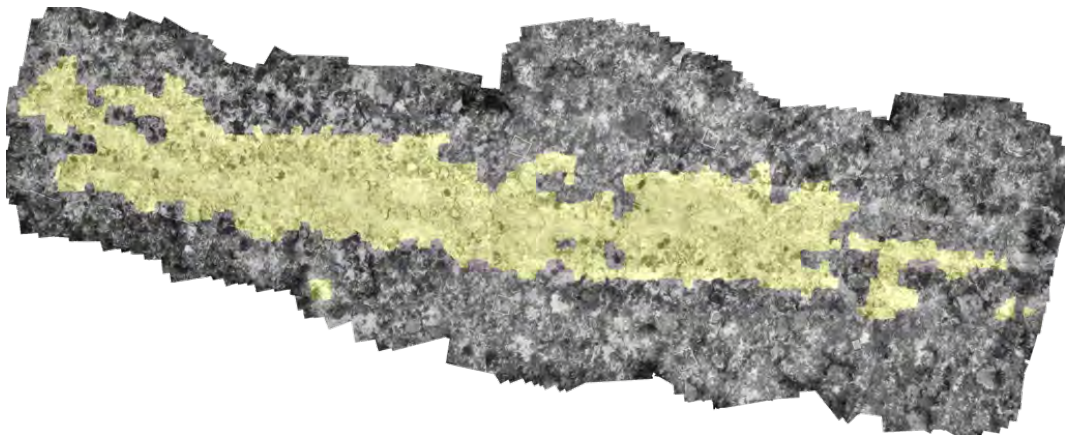
The configuration adaptation according to the characteristics of the datasets leads to better accuracy. Each dataset presents different challenges which can be better faced with proper tuning of the framework.

This chapter also illustrates that thematic mapping on a blended mosaic can provide better results than classifying individual images prior to creating the mosaic. When creating a mosaic from multiple overlapping images that have been classified individually, we will often have conflicting classification results for the same area of the mosaic. Currently the choice of the representative label for the mosaic is based on proximity to the image center, i.e., we choose the label of the image whose center is closest to a particular area in the mosaic. A simple extension of this approach, which will be done as future work, is to choose the label based on a voting scheme where the most represented label would be chosen as the final class on the mosaic.

5. RESULTS AND ANALYSIS



(a) Mosaic image

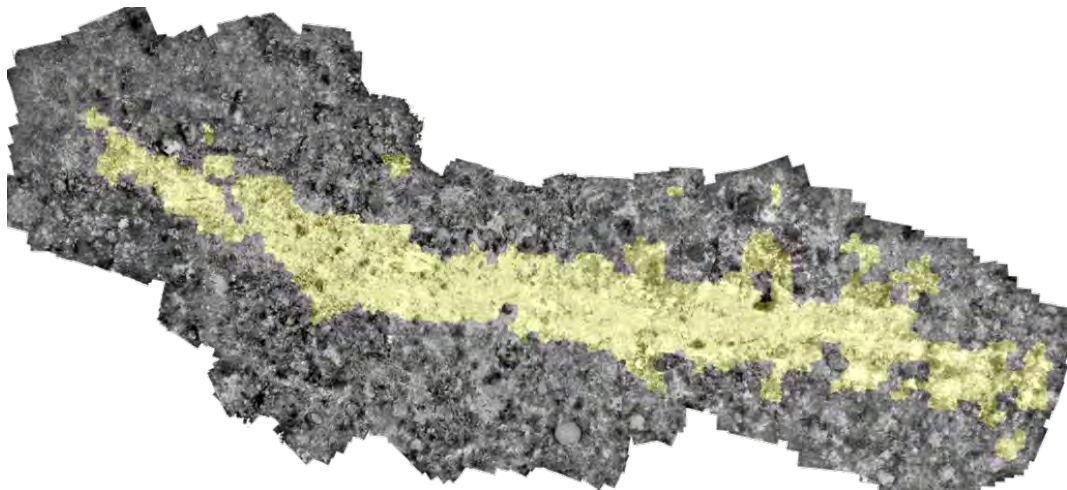


(b) Thematic mapping

Figure 5.14: Thematic mapping using *Grounding mosaic* of the year 2005.



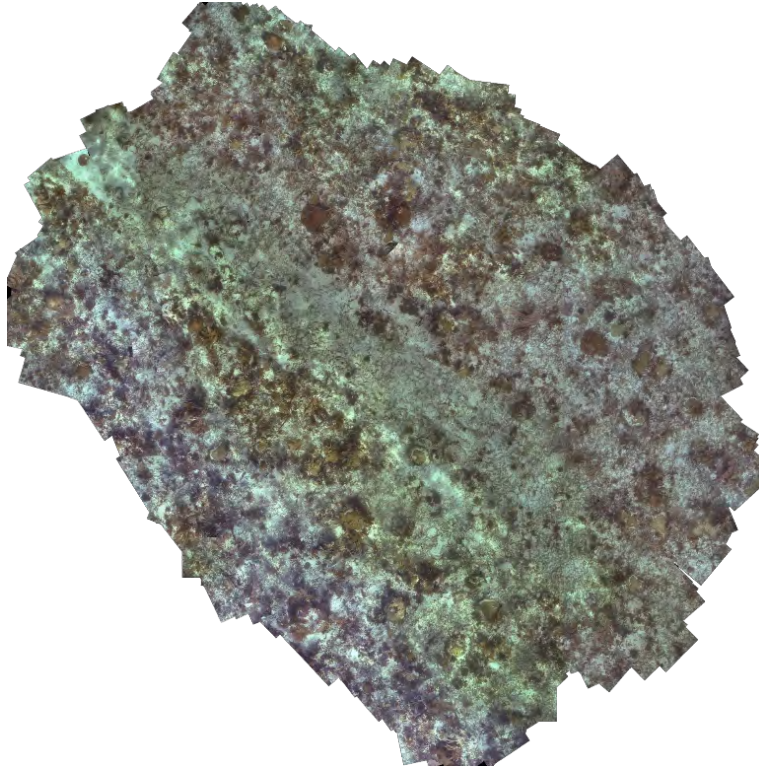
(a) Mosaic image



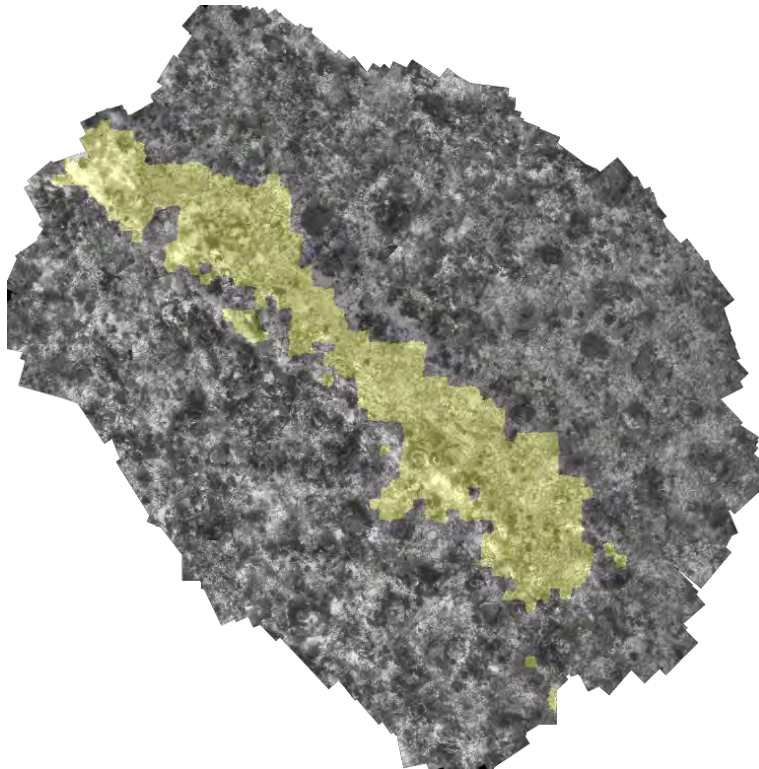
(b) Thematic mapping

Figure 5.15: Thematic mapping using *Grounding mosaic* of the year 2006.

5. RESULTS AND ANALYSIS



(a) Mosaic image



(b) Thematic mapping

Figure 5.16: Thematic mapping using *Grounding mosaic* of the year 2011.

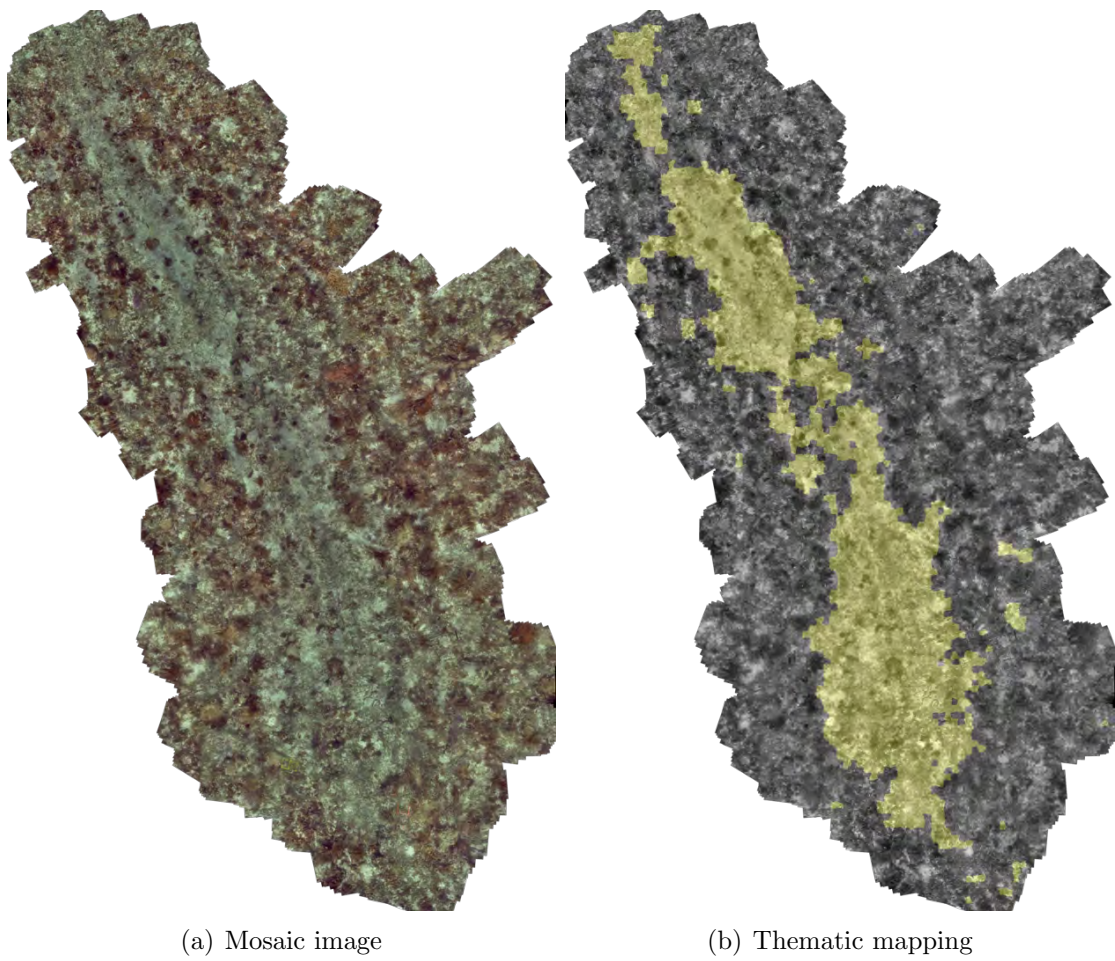


Figure 5.17: Thematic mapping using *Grounding mosaic* of the year 2012.

5. RESULTS AND ANALYSIS

Chapter 6

Conclusions

This chapter summarizes the results of our research in this thesis. The following sections provide an overview of the work carried out herein and discuss the achievements and contributions. The concluding section looks at some of the limitations of the method developed and also examines directions for future research.

6.1 Summary

The availability of large datasets of underwater images that have been enabled by recent advancements in digital acquisition systems and innovative platforms such as ROVs or AUVs provides new opportunities for remote sensing of seabed images as well as new challenges. The opportunity lies in the types of measurements that can be made from direct remote sensing of benthic organisms. The challenges are in efficiently extracting biological or ecological information from the raw images. Some form of the automated analysis will be required to make full use of these rich data sources.

Our proposed method presents a novel image classification scheme for underwater images that achieves the highest overall classification accuracy among any of the tested methods. In this thesis, ten challenging datasets were used for comparing the set of methods that are representative of the state of the art in automated classification of seabed images. On the latest standard benthic dataset '*Moorea labeled corals (MLC)*' [2], our method achieved 85.5% overall accuracy, whereas all

6. CONCLUSIONS

the other compared methods attained less than 80%, including Beijbom's method [2].

The proposed method can be configured to the characteristics (e.g., size, number of classes and resolution of the samples, color information availability, class types and so forth) of individual datasets. We provided guidelines for choosing the appropriate configuration for future classification of underwater images. The results suggest that a selective combination of various pre-processing steps, feature extraction and modification, prior probability of each class and choice of classifiers can provide better results than using a single method for all datasets. The results can be extended over large continuous areas by using mosaics of underwater images. On the *Red Sea mosaic*, our proposed method resulted in 83.7% overall accuracy, which is at least 8% higher than the other related methods tested under similar conditions.

The experimental part of our work allowed us to identify classification problems that are specific to underwater images. On one hand, there are many classes in underwater imagery that have samples with very clear differences in shape, color, texture, size, rotation, illumination, view angle, camera distance and light conditions. On the other hand, there are overlapping classes that look almost the same from specific angles and distances such as Crustose Coraline Algae (CCA) and Macroalgae (MA) in *MLC 2008* dataset. Finding an optimal patch size and patch shift are still open questions. Moreover, additional challenges, such as motion blurring, color attenuation, refracted sunlight patterns, water temperature variation, sky color variation and scattering effects on input images have to be addressed to maintain image quality and reliability of the information content. These issues highlight areas where future research may continue to improve the accuracy and efficiency of automated classification methods.

Applying an automated classification technique to mosaic composites produces a rapid (in terms of expert annotation time) technique of characterizing reef communities that can be used to track changes over time. Quantifying benthic community composition over the scale of hundreds of square meters by automated analysis of underwater image mosaics is a novel capability in ocean science and provides a new spatial scale from which sea bed dynamics can be observed and studied. Applying a trained classifier to a large-area mosaic image can reduce computational time since it bypasses the redundancy of classifying highly overlapping images. Furthermore,

the techniques presented here are not uniquely limited to ocean floor classification and may prove useful in other application scenarios with similarly high variability of content and image acquisition conditions.

The framework, experimental results and analysis presented in this thesis constitute a definite achievement regarding the goal of automated classification of underwater objects using optical imagery.

6.2 Main contributions

The work in this thesis allowed for a number of contributions. The main contributions are the following:

- Proposal of an image based underwater object classification framework which is configurable according to the characteristics of the dataset achieving state of the art results.
- Selection of appropriate approaches among those available in the Computer vision field for different steps and sub-steps of the proposed classification framework for underwater images.
- Development of an online sunflicker removal system using dynamic texture prediction and a motion compensated filtering approach. This method works with the help dynamic texture model learning and prediction based on a temporal or a 3D based median. This sunflicker removal method is one of the optional sub-steps in the pre-processing step of the proposed classification framework.
- Development of a framework for the creation of thematic maps from image composites (mosaics) which illustrates the advantages over classification on individual images.
- Investigating the possible use of 3D features together with 2D features for classification, specially in the case where the objects of interest (either man-made or natural) have particular shape or specific type of size or surface.

6. CONCLUSIONS

6.3 Limitations

The proposed method for classification contains the following limitations which will be addressed as future work.

- The proposed method is a supervised scheme, which means a certain amount of expert time is needed to manually annotate sufficient training examples for each class in the dataset. Although a large body of research has been devoted to unsupervised classification schemes, it is not clear if this approach would be of practical use for benthic ecologists and geologists, since these scientists tend to be quite specific on the particular classes that interest them.
- Our method needs a large number of examples from the background class (which belongs to none of the important classes in the dataset) to create a more accurate thematic map of the mosaic image. One of the main reasons behind this is that the background tends to have a very high variability in natural scenes. Therefore, a large background set is required to encompass that variability. However, this point can be considered as something unavoidable than a limitation.
- There can be size variations among the examples of any particular class. Using a fixed patch size can't always grasp the important texture features of that class due to large size variation. As a result, a fixed patch size may reduce the classification accuracy. Multi-resolution schemes should be explored in the future to cope with this limitation.
- When dealing with a new dataset, there is the need to tune a set of parameters to obtain a proper configuration that will perform the best. We need to adjust the proper configuration and parameters to get the best result. This means some information about the new dataset is required beforehand.

6.4 Future work

As future work, the following extensions to the proposed method will be addressed.

- Both the optical and 3D relief images created using optical structure from mo-

tion (SfM) techniques can be used together to acquire additional discriminative information about each class. Some underwater objects such as man-made objects (for example, military munitions) contain unique 3D shapes which are distinctive from natural objects on seabed. For such object class, SfM can produce additional discriminative shape features to help the classification process.

- A hierarchical class segmentation (from simple to complex) can provide more robustness to the method and can be explored in future work. In hierarchical class segmentation, initially objects are classified into broad category. Then sequentially, these broad categories are reclassified into more complex classes.
- Multi-modal classification methods that incorporate acoustic data, bathymetry, 3D terrain and magnetic data can definitely provide more information to the classifier to make more accurate decisions. With the recent advancements of underwater survey technologies, geo-referenced sensor data from different types of sensing modalities is available. Possible fusion of such data for better classification of the seabed can be a potential extension of our work.
- Detailed analysis of all the features used and their relation to the class types in the dataset can also be an area for research. The goal would be to avoid the need to perform dimension reduction, by using appropriate dataset-specific features.
- Incorporating large-scale variation in scale among the samples of the same classes could be performed by creating separate training sets of the same specific class with different sizes or using a fixed-size patch and representing images as multi-scale pyramids. In our method, all the features used are partially or completely scale and rotation invariant. These features are therefore able to mitigate the effects of limited scale variation of individual classes. For larger scale variation, it is important to have enough training examples of individual classes at different scales. In the future, multi-resolution mapping might be useful for benthic habitats covering large-scale variations of individual classes.
- An alternative to voting as future work could be to implement several classifiers for the same area (in this case we could even have more than one classifier in every image) and then use a random forest or boosting approaches to make the different classifiers predict a single class.

6. CONCLUSIONS

Appendix A

Datasets

The comparative performance of our proposed method with respect to the other leading methods has been tested on three standard texture datasets, five benthic datasets and three seabed mosaic datasets as described in this appendix. The datasets represent a variety of challenging environments. A brief summary of the image datasets used in this work is presented in Table A.1. Detailed descriptions of individual datasets together with visual examples are given in a later part of this appendix. The first four benthic datasets are used to guide selection of the various options available in our proposed framework. These four datasets are comprised of underwater images under varying light effects, color attenuation, scattering effects as well as many others. Testing the proposed methods on these datasets provides hints to select the appropriate options from these available ones for underwater imagery. Finally, datasets comprising three mosaics are used to illustrate the potential for generating thematic maps of large contiguous areas of the seabed using this automated classification framework.

A.1 Benthic datasets

Benthic datasets are created from seabed images acquired from different surveys. A detailed description with example images is given below.

A. DATASETS

Table A.1: A brief summary of the underwater image datasets used in this work for classification and thematic mapping. N represents number of patches in each dataset.

| Names | Classes | N | Resolution | Color | Type |
|-------------------------|---|-----------|------------|-------|---------|
| <i>EILAT</i> | Sand, urchin, branches type I, brain coral, favid coral, branches type II, dead coral and branches type III | 1,123 | 64 × 64 | Yes | Benthic |
| <i>RSMAS</i> | <i>Acropora cervicornis</i> , <i>Acropora palmata</i> , <i>Colpophyllia natans</i> , <i>Diadema antillarum</i> , <i>Diploria strigosa</i> , <i>Gorgonians</i> , <i>Millepora alcicornis</i> , <i>Montastraea cavernosa</i> , <i>Meandrina meandrites</i> , <i>Montipora spp.</i> , <i>Palythoa palythoa</i> , <i>Sponge fungus</i> , <i>Siderastrea siderea</i> and tunicates | 766 | 256 × 256 | Yes | Benthic |
| <i>EILAT 2</i> | Sand, urchin, branching coral, brain coral and favid coral | 303 | 128 × 128 | Yes | Benthic |
| <i>MLC 2008</i> | Crustose coralline algae (CCA), turf algae, macroalgae, sand, <i>Acropora</i> , <i>Pavona</i> , <i>Montipora</i> , <i>Pocillopora</i> , <i>Porites</i> | 18,872 | 312 × 312 | Yes | Benthic |
| <i>BMAT</i> | Bacterial mat, shell chaff, sand | 37,759 | 192 × 192 | No | Benthic |
| <i>UIUCtex</i> | Bark I, bark II, bark III, wood I, wood II, wood III, water, granite, marble, stone I, stone II, gravel, wall, brick I, brick II, glass I, glass II, carpet I, carpet II, fabric I, paper, fur, fabric II, fabric III and fabric IV | 1,000 | 640 × 480 | No | Texture |
| <i>CURET</i> | 61 texture materials imaged over varying pose and illumination, but at constant viewing distance. | 5,612 | 200 × 200 | No | Texture |
| <i>KTH-TIPS</i> | sandpaper, crumpled aluminum foil, Styrofoam, sponge, corduroy, linen, cotton, brown bread orange peel, cracker biscuit | 810 | 200 × 200 | No | Texture |
| <i>Red Sea mosaic</i> | Sand, urchin, branching coral, brain coral, favid coral, background objects | 73,600 | 64 × 64 | Yes | Mosaic |
| <i>North Sea mosaic</i> | Bacterial mat, Shell chaff, sand | 1,063,000 | 192 × 192 | Yes | Mosaic |
| <i>Grounding mosaic</i> | Natural reef, damaged reef | 13,762 | 180 × 180 | Yes | Mosaic |

A.1.1 *EILAT*

The *EILAT* dataset contains 1,123 image patches of 64×64 pixels (Figure A.1), taken from images of a reef survey near Eilat in the Red Sea [63]. A group of coral ecology experts have visually classified the images into eight classes (*sand*, *urchin*, *branch type I*, *brain coral*, *favid coral*, *branch type II*, *dead coral* and *branch type III*). Two of the classes have a larger number of examples compared with others. The image patches were extracted from the original full-size images, which were all taken with the same camera.


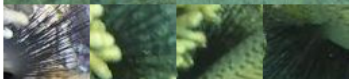
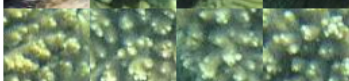





| Examples of patches | Class | Number | % in dataset |
|---|-------------|--------|--------------|
|  | Sand | 87 | 7.7% |
|  | Urchin | 80 | 7.1% |
|  | Branch I | 23 | 2.0% |
|  | Brain coral | 160 | 14.2% |
|  | Favid coral | 200 | 17.8% |
|  | Branch II | 216 | 19.2% |
|  | Dead coral | 280 | 24.9% |
|  | Branch III | 77 | 6.8% |

Figure A.1: Example images patches from the *EILAT* dataset showing 4 examples (in columns) of each of the eight classes (in rows, from top to bottom: sand, urchin, branch type I, brain coral, favid coral, branch type II, dead coral and branch type III).

A.1.2 *RSMAS*

The *RSMAS* dataset was obtained from reef survey images collected by divers from the Rosenstiel School of Marine and Atmospheric Sciences of the University of Miami. This dataset has examples of different classes of underwater coral reefs taken with different cameras at different times and places. This dataset consists of 766

A. DATASETS

image patches, each 256×256 pixels in size, of 14 different classes (see Figure A.2). The image patches cover a larger area on the ground than those in the *EILAT* dataset, which means they are more likely to contain mixed classes.















| Examples of patches | Class | Number | % in dataset |
|---|------------------------------|--------|--------------|
|  | <i>Acropora cervicornis</i> | 109 | 14.2% |
|  | <i>Acropora palmata</i> | 77 | 10.0% |
|  | <i>Colpophyllia natans</i> | 57 | 7.4% |
|  | <i>Diadema antillarum</i> | 63 | 8.2% |
|  | <i>Diploria Strigosa</i> | 24 | 3.1% |
|  | Gorgonians | 60 | 7.8% |
|  | <i>Millepora alcicornis</i> | 22 | 2.9% |
|  | <i>Montastraea cavernosa</i> | 79 | 10.3% |
|  | <i>Meandrina meandrites</i> | 54 | 7.1% |
|  | <i>Montipora</i> spp. | 28 | 3.7% |
|  | <i>Palythoa palythoa</i> | 32 | 4.2% |
|  | Sponge fungus | 88 | 11.5% |
|  | <i>Siderastrea siderea</i> | 37 | 4.8% |
|  | Tunicates | 36 | 4.7% |

Figure A.2: A subset of the *RSMAS* dataset showing 3 examples, in columns, of each of the 14 classes (in rows from top to bottom: *Acropora cervicornis* (ACER), *Acropora palmata* (APAL), *Colpophyllia natans* (CNAT), *Diadema antillarum* (DANT), *Diploria strigosa* (DSTR), *Gorgonians* (GORG), *Millepora alcicornis* (MALC), *Montastraea cavernosa* (MCAV), *Meandrina meandrites* (MMEA), *Montipora* spp. (MONT), *Palythoa palythoa* (PALY), *Sponge fungus* (SPO), *Siderastrea siderea* (SSID) and tunicates (TUNI).

A.1.3 *MLC 2008*

The *MLC 2008* dataset [2] is a subset of images collected for the Moorea-Labeled Corals (MLC) from the Moorea Coral Reef Long Term Ecological Research site (MCR-LTER) packaged for computer vision research and is the only natural habitat dataset that was not created specifically for the work of this thesis. The Moorea-labeled corals (MLC) dataset was published by Beijbom *et al.* [2] in 2012. It contains 2,055 images from three habitats: fringing reef, outer 10 m and outer 17 m, acquired in 2008, 2009 and 2010. It also contains random points that have been annotated. The nine most abundant labels include four non-coral classes: (1) Crustose Coralline algae (CCA), (2) turf algae (Turf), (3) macroalgae (Macro) and (4) sand; and five coral genera: (5) *Acropora* (Acrop), (6) *Pavona* (Pavon), (7) *Montipora* (Monti), (8) *Pocillopora* (Pocill) and (9) *Porites* (Porit). For our work, we use a subsection of this MLC dataset which is mainly made up of the images collected in 2008. From the *MLC 2008* dataset, we randomly selected 18,872 image patches of 312×312 pixels centered on the annotated points. All nine classes were represented in the random image patches (Figure A.3).

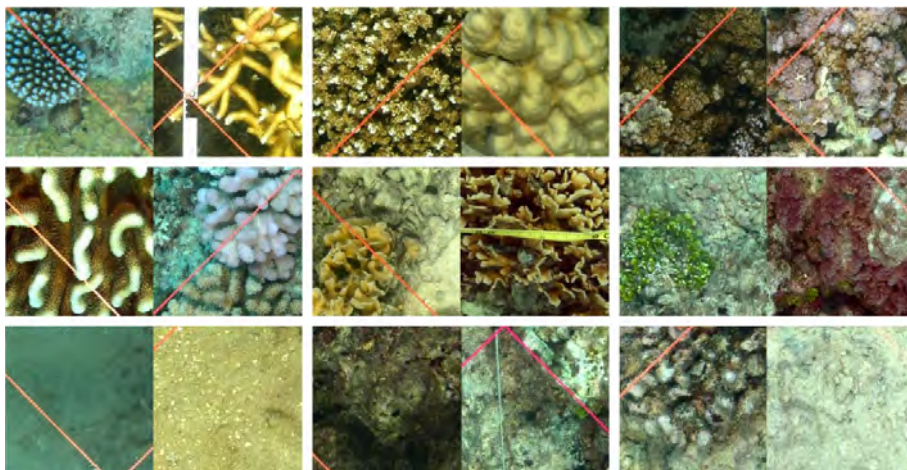


Figure A.3: A subset of the *MLC 2008* dataset showing two examples for each class. First row: *Acropora*, *Porites*, *Montipora*. Second row: *Pocillopora*, *Pavona*, Macroalgae. Third row: sand, turf algae, Crustose Coralline algae (CCA).

A. DATASETS

A.1.4 *EILAT 2*

The *EILAT 2* dataset contains 303 image patches. A group of experts has visually classified the images into five classes: sand, urchin, branching coral, brain coral and favid coral (Figure A.4). The patches are of medium resolution, 128×128 pixels, taken from points on the object to keep the visual aspects of the object and, in some cases, a portion of the background. Here, all the images have been taken with the same camera.






| Examples of patches | Class | Number | % in dataset |
|---|-----------------|--------|--------------|
|  | Favid coral | 89 | 29.4% |
|  | Brain coral | 71 | 23.4% |
|  | Branching coral | 49 | 16.2% |
|  | Sand | 80 | 26.4% |
|  | Urchin | 14 | 4.6% |

Figure A.4: A subset of the *EILAT 2* dataset showing 4 examples in columns of each of the five classes (in rows, from top to bottom: favid coral, brain coral, branching coral, sand and urchin).

A.1.5 *BMAT*

The *BMAT* dataset was created from images of a large area survey using a Hugin AUV [41] in the North Sea. The survey was done under the EU/FP7 ECO2 project, where they covered the area overlying the subsurface CO_2 storage site of the Utsira formation around the Sleipner platforms [95]. One of the main goals of these expeditions was to test techniques for high-resolution optical imaging to create photo-mosaics and map bacterial mats on the seafloor associated with fluid outflow. Identification of bacterial mats and their spatial distributions may guide studies to determine if associated fluid outflows are linked to CO_2 escapes or to natural processes. The main classes present in these images are *bacterial mats*, *shell chaff* and *sand*. The *BMAT* dataset comprises almost 37,759 images. Patches of 192 by 192 pixels are extracted from the cropped survey images to be manually annotated by experts.

Figure A.5 shows examples of a single patch from each class. Experts manually classified 24,449 patches (9,625 patches of sand, 14,427 patches of shell chaff and 447 patches of bacterial mat) representing 2.3% of the total number of the image patches captured during the tenth dive.

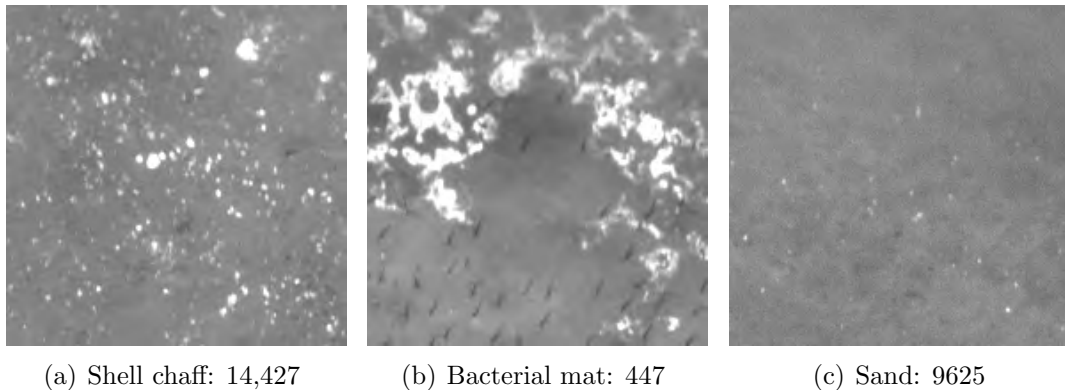


Figure A.5: Examples of training patches of each individual class (shell chaff, bacterial mat, sand) of the *BMAT* dataset. Each patch is 192 by 192 pixels covering an area of approximately 0.6 square meters. Number of patches is mentioned in each figure of that particular class.

A.2 Texture datasets

These are standard texture datasets created by different groups in the computer vision community for testing classification algorithms. In our work, we used three of the most popular as described below.

A.2.1 *UIUCtex*

The University of Illinois at Urbana-Champaign texture (*UIUCtex*) dataset [56] contains 40 images in each of 25 texture classes: *bark I*, *bark II*, *bark III*, *wood I*, *wood II*, *wood III*, *water*, *granite*, *marble*, *stone I*, *stone II*, *gravel*, *wall*, *brick I*, *brick II*, *glass I*, *glass II*, *carpet I*, *carpet II*, *fabric I*, *paper*, *fur*, *fabric II*, *fabric III* and *fabric IV* (represented in Figure A.6). These textures were viewed under significant scale and viewpoint variation. The dataset includes non-rigid deformations, illumination changes and viewpoint-dependent appearance variations. All the image patches in this dataset are of 640×480 pixels.

A. DATASETS

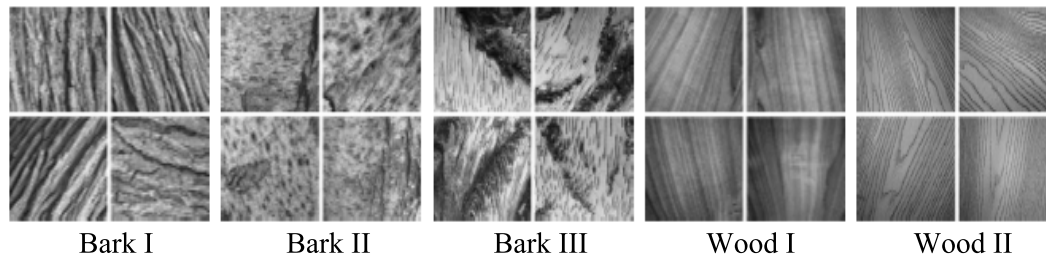


Figure A.6: A subset of the *UIUCtex* dataset showing 4 examples from 5 of the 25 classes (from the left group of 4 to the right: bark I, bark II, bark III, wood I and wood II).

A.2.2 *CURET*

The Columbia-Utrecht Reflectance and Texture (*CURET*) dataset [17] contains 61 texture classes, each with 92 images of 200×200 pixels. The materials were imaged with varying positions and illumination, but at a constant viewing distance. The changes of viewpoint and illumination direction significantly affect the texture's appearance (Figure A.7).

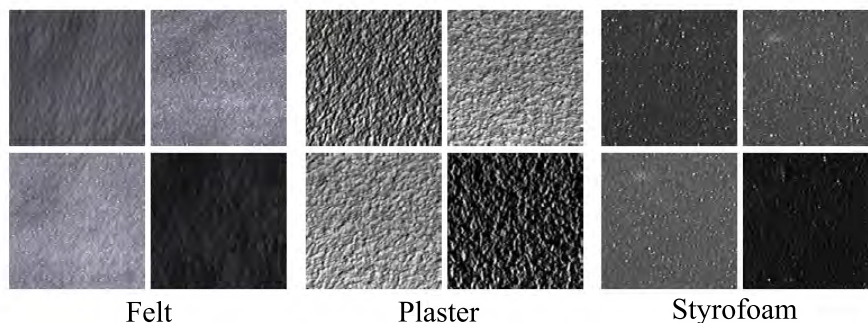


Figure A.7: A subset of the *CURET* texture dataset showing 4 examples from each of the three classes (from the left group of four to the right: felt, plaster and Styrofoam). Note the large intra-class variability caused by viewpoint and illumination changes.

A.2.3 *KTH-TIPS*

The Kungliga Tekniska Hogskolan Textures under varying Illumination, pose and Scale (*KTH-TIPS*) dataset [6] (illustrated in Figure A.8) contains images of 10 types of natural materials and provide variations in scale, in pose and illumination. The images were captured at nine scales spanning two octaves (relative scale changes from 0.5 to 2), viewed under three different illumination directions and three different

poses, thus giving a total of nine images per scale and 81 images per material of 200×200 pixels. In total, there are 810 images comprising 10 different classes (sandpaper, crumpled aluminum foil, Styrofoam, sponge, corduroy, linen, cotton, brown bread, orange peel and cracker biscuit).

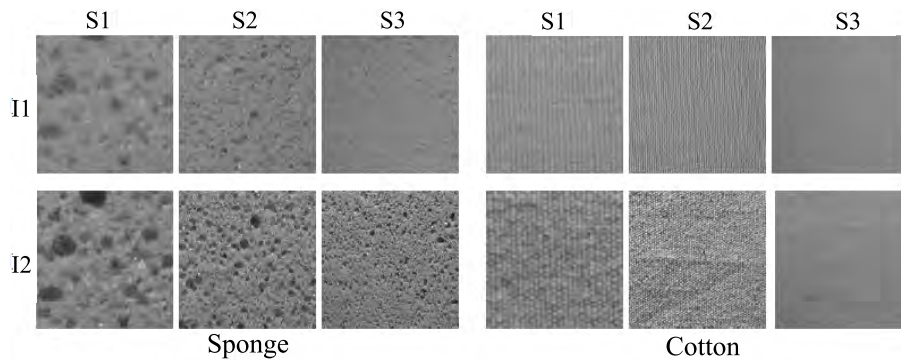


Figure A.8: A subset of the *KTH-TIPS* dataset showing six examples from 2 of the 10 classes: sponge (left) and cotton (right). The six examples include two different illuminations and three different scales.

A.3 Mosaic datasets

In this section, we describe the three mosaic image datasets that were created to evaluate the thematic mapping performance of our proposed method.

A.3.1 The *Red Sea mosaic*

The *Red Sea mosaic* (Figure A.9) was created using the methods described in [23; 59; 60; 83] from diver survey images. The *Red Sea mosaic* comprises 283 high-resolution digital still color images rendered at 1 mm per pixel of shallow water coral reefs in the Red Sea, near Eilat [63]. The mosaic covers an area of 19.2 square meters (3256×5937 pixels with color information). To extract patches, a sliding window method was used. The sliding window was moved with 16 pixel shifts, both horizontally and vertically. In total, 73,600 image patches were created from the mosaic image. Neighboring patches are 64×64 pixels and have an overlap of 75% maximum.

A. DATASETS

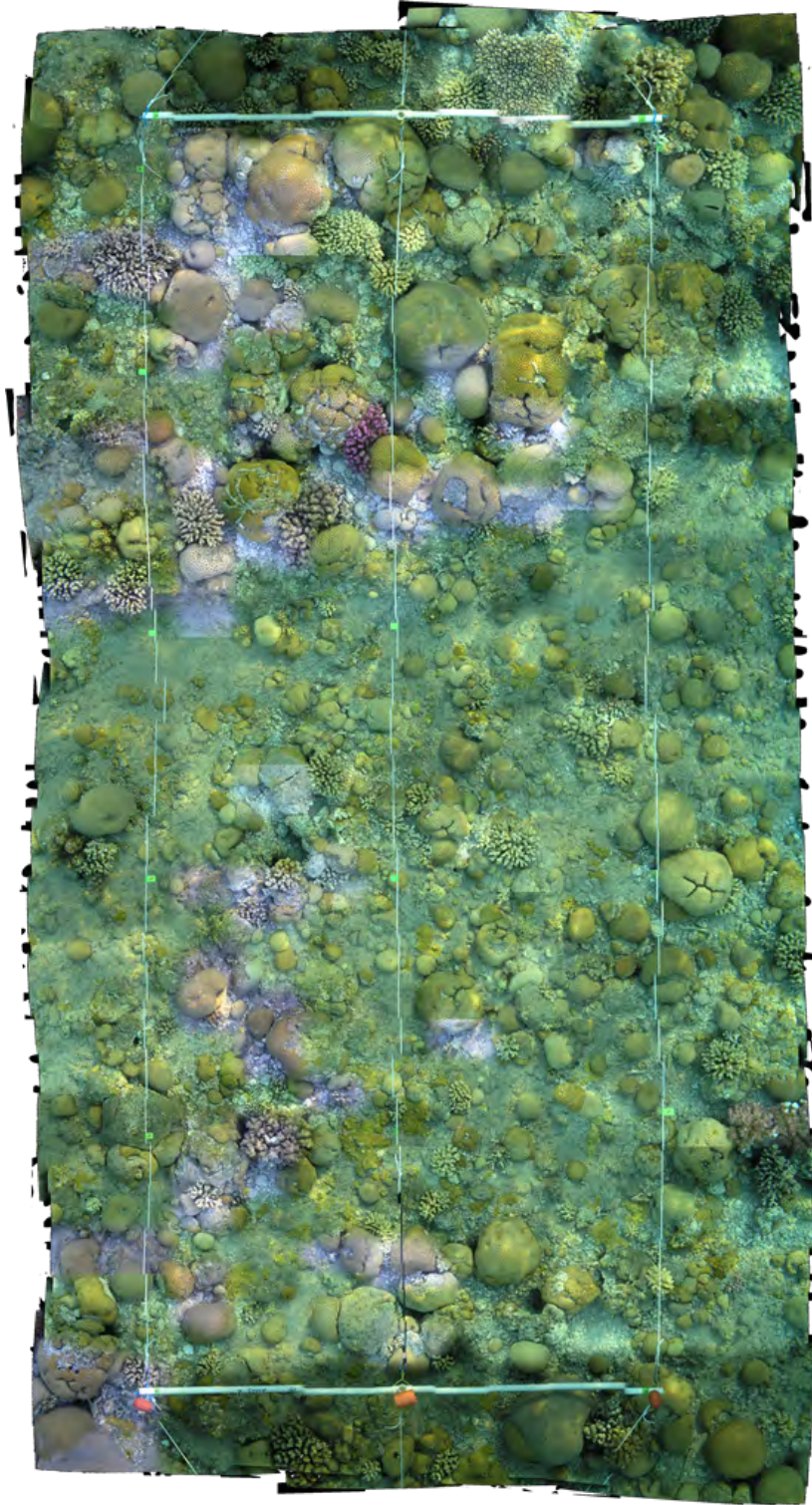


Figure A.9: The *Red Sea mosaic* image covering an area of 19.2 square meters ($3,256 \times 5,937$ pixels).

A.3.2 The *North Sea mosaic*

The *North Sea mosaic* was created from images of a survey over the area overlying the subsurface CO_2 storage site of the Utsira formation around the Sleipner platforms [41]. A Hugin AUV was used to collect a total of 37,759 grey images corresponding to a single dive survey. Initially, the raw survey images were cropped to 1,024 by 256 pixels to the borders of the image which were blurry and noisy due to the uneven artificial illumination used in the AUV. *The North Sea mosaic* (Figure A.10) was created from a subset of 340 images using the same methods as the *Red Sea mosaic*. The mosaic covers an area of 232.86 square meters with a width of 7.38 meters and a height of 31.5738 meters.

A.3.3 The *Grounding mosaic*

On December 5, 2002, the 49-foot vessel Evening Star ran aground on a hard bottom community dominated by stony and soft corals in the waters of Biscayne National Park, Florida (2523.332 N, 8009.874 W, 3 m of depth). On May 23, 2005, video data of the damaged and surrounding areas was collected using a Sony TRV900 DV camcorder placed in an underwater housing [59]. The *Grounding mosaic* was created from 2,149 video frames with the presence of sunflicker and from those same images a second mosaic was created having sunflicker removed. The minimum final ground resolution is 5 mm/pixel. The area of the mosaic is 291 square meters. the injury is documented to be 45 m in length and ranged from 3 to 5 m in width. The *Grounding mosaic* (Figure A.10) was created using the same methods as with the *Red Sea mosaic*. The sunflicker was removed from the video frames using a sunflicker removal method with an offline [37] approach.

Also other *Grounding mosaics* are created from the images from 2005, 2006, 2011 and 2012 surveys respectively. These mosaics are used to quantify change detection among coral reef distribution on the seabed along the years specially where the ship wrecking happened.

A. DATASETS

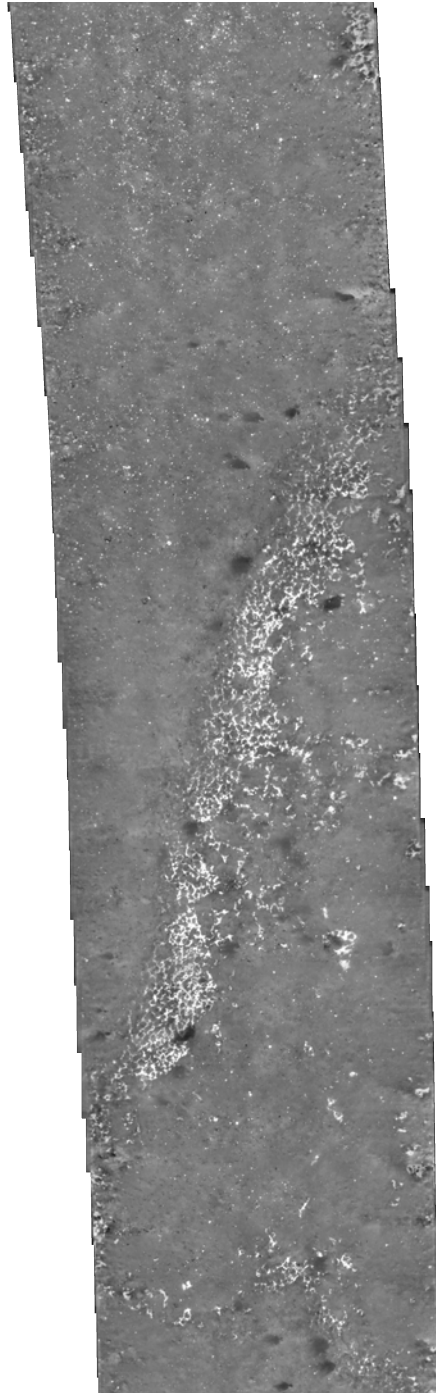
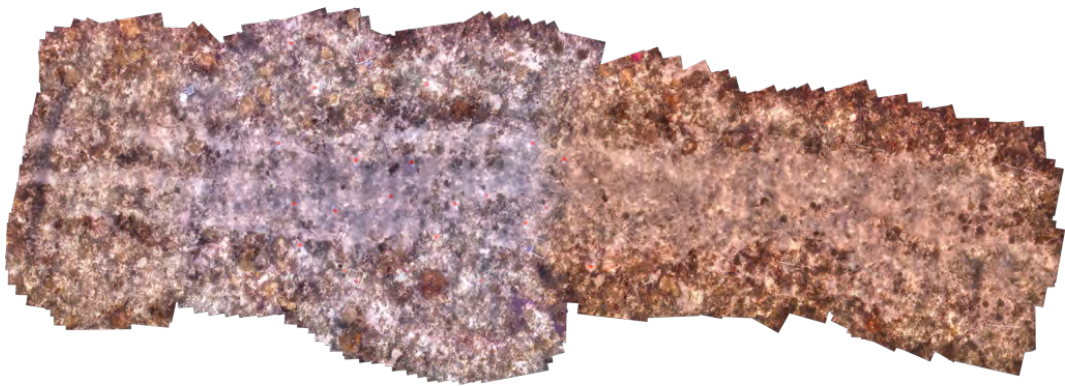
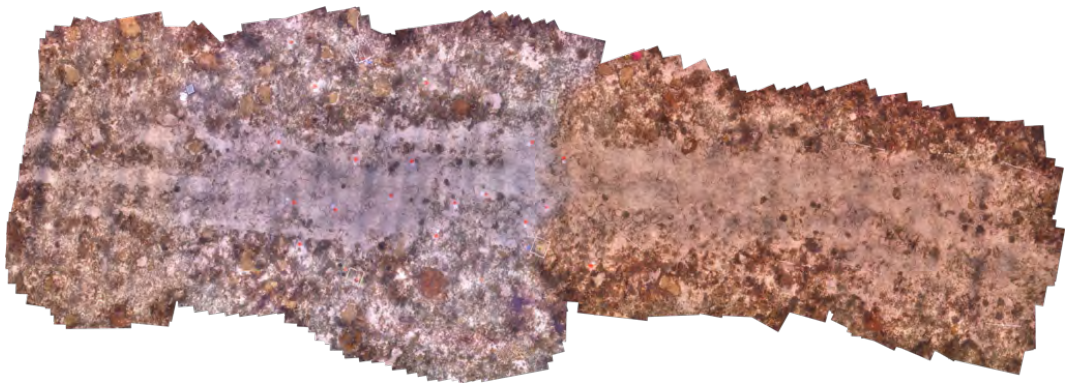


Figure A.10: The *North Sea mosaic* image covering an area of 232.86 square meters with a width of 7.38 meters and a height of 31.5738 meters ($1,821 \times 6,652$ pixels).



(a) Mosaic creation with sunflicker effects present



(b) Mosaic creation with sunflicker effects removed

Figure A.11: The *Grounding mosaic* image covering an area of 291 square meters illustrating an injury of width ranging from 3 to 5 meters and a length of 45 meters ($4,200 \times 11,796$ pixels). Here first mosaic is created having the sunflicker present and the 2nd mosaic is created after removing sunflicker effect.

A.4 *Ordnance dataset*

The *Ordnance* dataset is mainly used to test the performance of the 3D features. This dataset has mainly two main classes, bomb shell and background. The Bomb shells tend to have a distinctive 3D shape that can be detected with better accuracy using 3D cues. This dataset was collected by the National Oceanic and Atmospheric Administration (NOAA) at a site called 'Ordnance Reef' off of Waianae, Hawaii by divers using a hand-held HDV video camera. The camera was a Sony HDR-XR520V which did not have progressive scan, so images were cut in half to deal with the interlacing. The resolution of the *Ordnance mosaic* (Figure A.12) is $7,949 \times 8,444$ pixels. Figure A.12 illustrates the mosaic of the Ordnance dataset. Figure A.13 illustrates the zoomed in version of the three markers on the figure A.12 of *Ordnance* mosaic. This dataset is very challenging as the corals grow over the bomb shells, thus matching both types of objects having similar textures on their surfaces..

A.5 Summary

This chapter describes the datasets used for evaluating and comparing methods. The characteristics of the datasets (e.g., the size of the dataset, number of classes, resolution of the samples, color information availability, etc.) is very important information for our proposed method as this method is configurable according to it. The three standard texture datasets are used to evaluate our method with available state-of-the-art texture classification algorithm in the computer vision field.

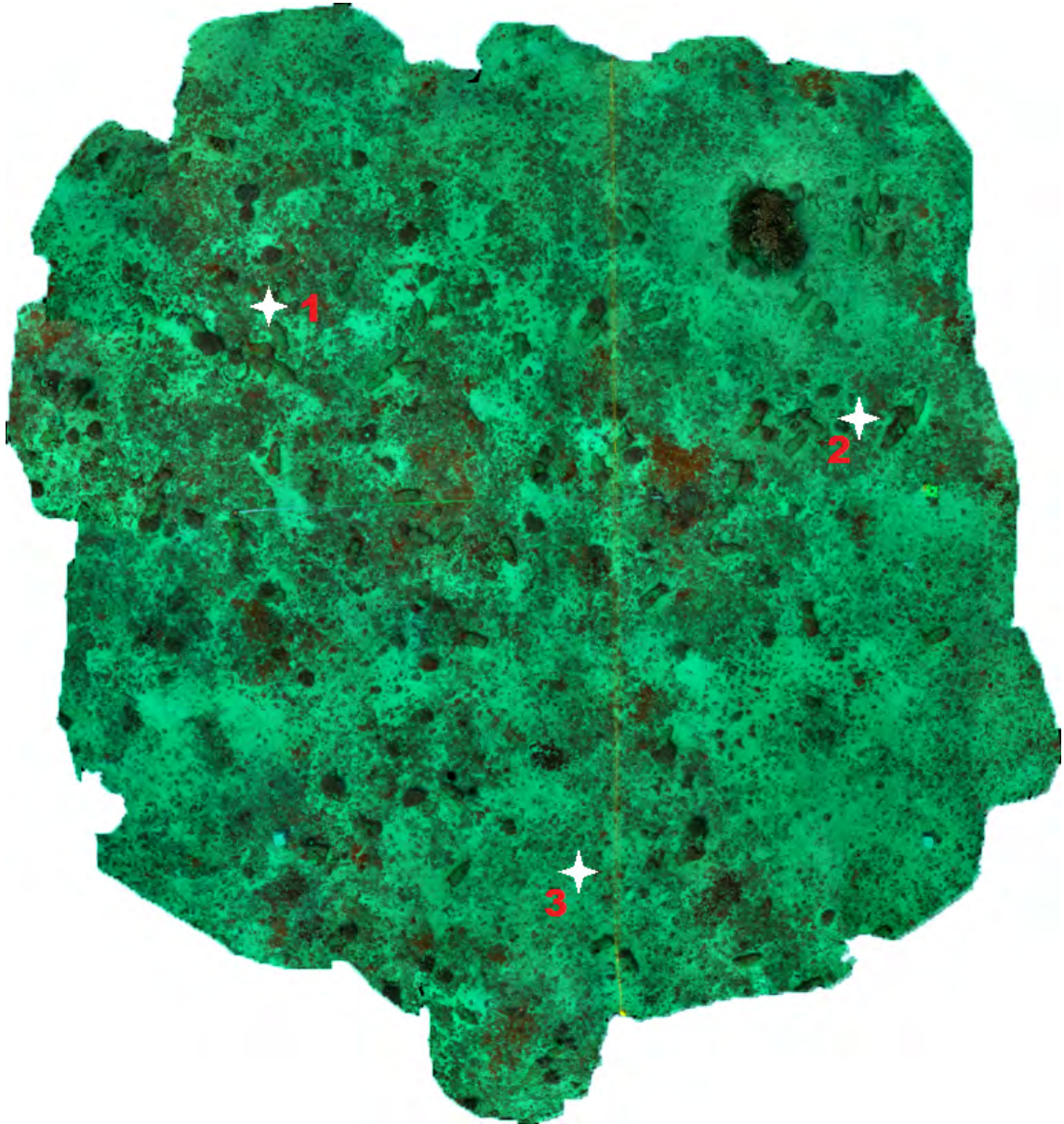
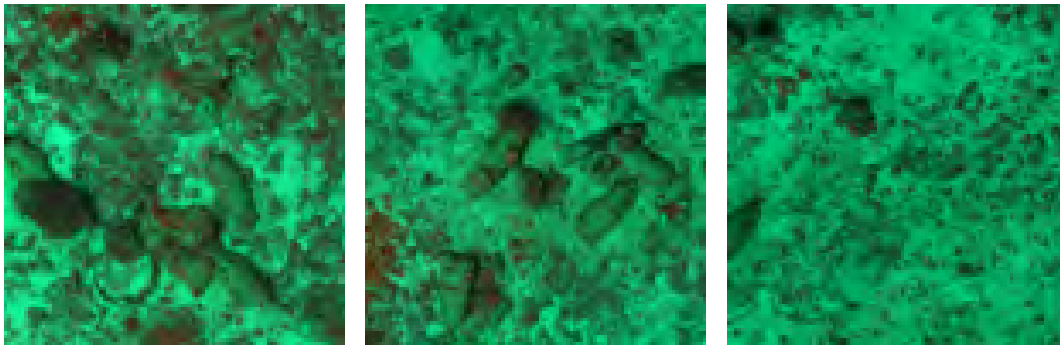


Figure A.12: The *Ordnance mosaic* image having the resolution of $7,949 \times 8,444$ pixels.

A. DATASETS



(a) Marker 1

(b) Marker 2

(c) Marker 3

Figure A.13: Examples of small parts of the *Ordnance* mosaic. These 3 patches corresponds to zoomed in version of the three markers on the Figure A.12

References

- [1] R. ARMSTRONG, H. SINGH, J. TORRES, R. NEMETH, A. CAN, C. ROMAN, R. EUSTICE, L. RIGGS, AND G. GARCIA-MOLINER. Characterizing the deep insular shelf coral reef habitat of the hind bank marine conservation district using the seabed autonomous underwater vehicle. *Continental Shelf Research*, **26**[2]:194 – 205, 2006. 3
- [2] O. BEIJBOM, P. EDMUNDS, D. KLINE, B. MITCHELL, AND D. KRIEGMAN. Automated annotation of coral reef survey images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, Rhode Island, 2012. x, 4, 13, 14, 17, 19, 24, 49, 93, 94, 95, 117, 118, 127
- [3] A. BENDER, S. WILLIAMS, AND O. PIZARRO. Classification with probabilistic targets. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1780–1786, 2012. 13
- [4] M. BEWLEY, B. DOUILLARD, N. NOURANI-VATANI, A. FRIEDMAN, O. PIZARRO, AND S. WILLIAMS. Automated species detection: An experimental approach to kelp detection from sea-floor auv images. 2012. 13, 14
- [5] A. BOVIK, M. CLARK, AND W. GEISLER. Multichannel texture analysis using localized spatial filters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **12**[1]:55 –73, Jan 1990. 57
- [6] B. CAPUTO, E. HAYMAN, M. FRITZ, AND J. EKLUNDH. Classifying materials in the real world. *Image and Vision Computing*, **28**[1]:150–163, January 2010. 130

REFERENCES

- [7] V. CASELLES. Geometric models for active contours. In *Image Processing, 1995. Proceedings., International Conference on*, **3**, pages 9–12 vol.3, 1995. 54
- [8] D. CLAUSI. An analysis of co-occurrence texture statistics as a function of grey level quantization. *Canadian Journal of Remote Sensing*, **28**[1]:45–62, 2002. 59, 60, 80
- [9] R. CLEMENT, M. DUNBABIN, AND G. WYETH. Toward robust image detection of crown-of-thorns starfish for autonomous population monitoring. In *Australasian Conference on Robotics and Automation*, 2005. 11, 14
- [10] J. COHEN. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, **20**[1]:37–46, April 1960. 20, 21
- [11] R. CONGALTON. A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sensing of Environment*, **37**[1]:35–46, July 1991. 81
- [12] C. CORTES AND V. VAPNIK. Support-vector networks. *Machine Learning*, **20**[3]:273–297, 1995. 11, 13, 76, 80
- [13] P. COUTO. Assessing the accuracy of spatial simulation models. *Ecological Modelling*, **167**[2]:181 – 198, 2003. 19, 20, 24
- [14] T. COVER AND P. HART. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, **13**[1]:21–27, 1967. 75, 80
- [15] G. CSURKA, C. DANCE, L. FAN, J. WILLAMOWSKI, AND C. BRAY. Visual categorization with bags of keypoints. *Workshop on statistical learning in computer vision, ECCV*, **1**:22, 2004. 12, 16
- [16] P. CULVERHOUSE, R. WILLIAMS, B. REGUERA, V. HERRY, AND S. GONZALEZ-GIL. Do experts make mistakes? a comparison of human and machine identification of dinoflagellates. *Marine Ecology Progress Series*, **247**:17–25, 2003. 5
- [17] K. DANA, B. VAN GINNEKEN, S. NAYAR, AND J. KOENDERINK. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics. TOG*, **18**[1]:1–34, January 1999. 130

REFERENCES

- [18] J. DAVID. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, **4**:2379–2394, 1987. 70
- [19] J. DIAZ AND R. TORRES. Classification of underwater color images with applications in the study of deep coral reefs. In *Circuits and Systems, 2006. MWSCAS '06. 49th IEEE International Midwest Symposium on*, **2**, pages 610–613, 2006. 12
- [20] G. DORETTO, A. CHIUSO, Y. WU, AND S. SOATTO. Dynamic textures. *International Journal of Computer Vision. IJCV 03*, **51**:91–109, 2003. 25, 28, 35
- [21] J. EMMANUEL, X. LI, Y. MA, AND J. WRIGHT. Robust principal component analysis. *Computing Research Repository. CoRR*, **abs/0912.3599**, 2009. 28
- [22] J. ESCARTIN, R. GARCIA, T. BARREYRE, M. CANNAT, N. GRACIAS, A. SHIHAVUDDIN, AND E. MITTELSTAEDT. Optical methods to monitor temporal changes at the seafloor: The lucky strike deep-sea hydrothermal vent field (mid-atlantic ridge). In *Underwater Technology Symposium (UT), 2013 IEEE International*, pages 1–6, March 2013. i
- [23] J. ESCARTIN, R. GARCIA, O. DELAUNOY, J. FERRER, N. GRACIAS, A. ELIBOL, X. CUFI, L. NEUMANN, D. FORNARI, S. HUMPHRIS, AND J. RENARD. Globally aligned photomosaic of the lucky strike hydrothermal vent field: Release of georeferenced data, mosaic construction, and viewing software. *Geochemistry, Geophysics, Geosystems*, **9**[12], 2008. 77, 103, 131
- [24] D. FINLAYSON, S. DREW, AND V. BRIAN. Color constancy: generalized diagonal transforms suffice. *Journal of the Optical Society of America A*, **11**[11]:3011–3019, Nov 1994. 64
- [25] G. FINLAYSON, B. SCHIELE, AND J. CROWLEY. Comprehensive colour image normalization. In *European Conference on Computer Vision, ECCV*, pages 475–490, 1998. 63
- [26] J. FINN. Use of the average mutual information index in evaluating classification error and consistency. *International Journal of Geographical Information Systems*, **7**[4]:349–366, 1993. 20

REFERENCES

- [27] M. FISCHLER AND R. BOLLES. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the Association for Computing Machinery. CACM*, **24**[6]:381–395, June 1981. 31
- [28] G. FOODY. Thematic map comparison: Evaluating the statistical significance of differences in classification accuracy. *Photogrammetric engineering and remote sensing*, **70**[5]:627–633, 2004. 19, 24
- [29] J. FRANCISCO, E. RAUL, AND O. LUIS. Hough transform for robust segmentation of underwater multispectral images. pages 591–600, 2003. 12
- [30] A. FRIEDMAN, O. PIZARRO, AND S. WILLIAMS. Rugosity, slope and aspect from bathymetric stereo image reconstructions. In *OCEANS 2010 IEEE - Sydney*, pages 1–9, 2010. 64, 69
- [31] A. FRIEDMAN, O. PIZARRO, S. WILLIAMS, AND M. JOHNSON-ROBERSON. Multi-scale measures of rugosity, slope and aspect from benthic stereo image reconstructions. *PLoS ONE*, **7**[12]:e50440, 2012. 13, 15
- [32] B. FULKERSON, A. VEDALDI, AND S. SOATTO. Class segmentation and object localization with superpixel neighborhoods. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 670–677, 2009. 77
- [33] R. GARCIA, T. NICOSEVICI, AND X. CUFI. On the way to solve lighting problems in underwater imaging. In *OCEANS '02 MTS/IEEE*, **2**, pages 1018–1024 vol.2, 2002. 52
- [34] P. GEHLER AND S. NOWOZIN. On feature combination for multiclass object classification. In *2009 IEEE 12th International Conference on Computer Vision. ICCV*, pages 221 –228, 29 2009-oct. 2 2009. 56
- [35] A. GLEASON, R. REID, AND K. VOSS. Automated classification of underwater multispectral imagery for coral reef monitoring. In *OCEANS IEEE/MTS 2007, Vancouver*, pages 1 –8, 29 2007-Oct. 4 2007. 11, 14
- [36] N. GRACIAS, M. MAHOOR, S. NEGAHDARIPOUR, AND A. GLEASON. Fast image blending using watersheds and graph cuts. *Image and Vision Computing*, **27**[5]:597 – 607, 2009. The 17th British Machine Vision Conference (BMVC 2006). 77

REFERENCES

- [37] N. GRACIAS, S. NEGAHDARIPOUR, L. NEUMANN, R. PRADOS, AND R. GARCIA. A motion compensated filtering approach to remove sunlight flicker in shallow water images. In *OCEANS'08 MTS/IEEE Conference, Quebec*, sept. 2008. v, 5, 25, 28, 31, 33, 40, 41, 46, 54, 133
- [38] N. GRACIAS, S. VAN DER ZWAAN, A. BERNARDINO, AND J. SANTOS-VICTOR. Mosaic-based navigation for autonomous underwater vehicles. *Oceanic Engineering, IEEE Journal of*, **28**[4]:609–624, Oct 2003. 64
- [39] S. GRIGORESCU, N. PETKOV, AND P. KRUIZINGA. Comparison of texture features based on gabor filters. *Image Processing, IEEE Transactions on*, **11**[10]:1160 – 1167, oct 2002. 57
- [40] Z. GUO, L. ZHANG, AND D. ZHANG. A completed modeling of local binary pattern operator for texture classification. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **19**[6]:1657 –1663, june 2010. 59, 80
- [41] P. HAGEN AND J. KRISTENSEN. The hugin auv "plug and play" payload system. In *OCEANS '02 MTS/IEEE*, **1**, pages 156–161 vol.1, 2002. 128, 133
- [42] G. HALEY AND B. MANJUNATH. Rotation-invariant texture classification using modified gabor filters. In *Image Processing, 1995. Proceedings., International Conference on*, **1**, pages 262 –265 vol.1, oct 1995. 57
- [43] R. HARALICK, K. SHANMUGAM, AND I. DINSTEN. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, **SMC-3**[6]:610 –621, Nov. 1973. 10, 58, 59, 60, 80
- [44] E. HAYMAN, B. CAPUTO, M. FRITZ, AND J. EKLUNDH. On the significance of real-world conditions for material classification. In *European Conference on Computer Vision. ICCV*, pages 253–266, 2004. 15, 95
- [45] T. HESKES. Energy functions for self-organizing maps. In E. OJA AND S. KASKI, editors, *Kohonen Maps*, pages 303–315. Elsevier Science Inc., Amsterdam, 1999. 10
- [46] H. HOFFMANN. Kernel pca for novelty detection. *Pattern Recognition*, **40**[3]:863–874, March 2007. 12

REFERENCES

- [47] M. JOHNSON-ROBERSON, S. KUMAR, O. PIZARRO, AND S. WILLAMS. Stereoscopic imaging for coral segmentation and classification. In *OCEANS IEEE/MTS 2006, Boston*, pages 1–6, Sept. 2006. 11, 14
- [48] M. JOHNSON-ROBERSON, S. KUMAR, AND S. WILLAMS. Segmentation and classification of coral for oceanographic surveys: A semi-supervised machine learning approach. In *OCEANS IEEE/MTS 2006, Asia Pacific*, May 2006. 11, 14
- [49] D. KEREN AND C. GOTSMAN. Fitting curves and surfaces with constrained implicit polynomials. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **21**[1]:31–41, Jan 1999. 66
- [50] M. KHALID AND M. BILAL MALIK. Biased nature of bhattacharyya coefficient in correlation of gray-scale objects. In *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, pages 209–214, 2005. 72
- [51] S. KICHENASSAMY, A. KUMAR, P. OLVER, A. TANNENBAUM, AND A. YEZZI. Gradient flows and geometric active contour models. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 810–815, 1995. 54
- [52] J. KNOPP, M. PRASAD, G. WILLEMS, R. TIMOFTE, AND L. VAN GOOL. Hough transform and 3d surf for robust three dimensional classification. In KOSTAS DANIILIDIS, PETROS MARAGOS, AND NIKOS PARAGIOS, editors, *Proceedings of the European Conference on Computer Vision*, Lecture Notes in Computer Science, pages 589–602. Springer, September 2010. 15
- [53] R. KOHAVI. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, International Joint Conferences on Artificial Intelligence. IJCAI’95, pages 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. 82
- [54] P. KOVESI. Symmetry and asymmetry from local phase. In *Tenth Australian Joint Conference on Artificial Intelligence*, pages 2–4, 1997. 68, 70

REFERENCES

- [55] P. KOVESI. Matlab and octave functions for computer vision and image processing. *School of Computer Science Software Engineering*, 2009. 36
- [56] S. LAZEBNIK, C. SCHMID, AND J. PONCE. A sparse texture representation using local affine regions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **27**[8]:1265–1278, aug. 2005. 129
- [57] A. LEVINSHTEIN, A. STERE, K. KUTULAKOS, D. FLEET, S. DICKINSON, AND K. SIDDIQI. Turbopixels: Fast superpixels using geometric flows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **31**[12]:2290–2297, 2009. 54
- [58] S. LIAO, M. LAW, AND A. CHUNG. Dominant local binary patterns for texture classification. *Image Processing, IEEE Transactions on*, **18**[5]:1107–1118, 2009. 63
- [59] D. LIRMAN, N. GRACIAS, B. GINTERT, A. GLEASON, G. DEANGELO, M. GONZALEZ, E. MARTINEZ, AND R. REID. Damage and recovery assessment of vessel grounding injuries on coral reef habitats using georeferenced landscape video mosaics. *Limnology and Oceanography: Methods*, **8**:88–97, 2010. 77, 103, 131, 133
- [60] D. LIRMAN, N. GRACIAS, B. GINTERT, A. GLEASON, R. REID, S. NEGAHDARIPOUR, AND P. KRAMER. Development and application of a video-mosaic survey technology to document the status of coral reef communities. *Environmental Monitoring and Assessment*, **125**[1-3]:59–73, 2007. 64, 77, 103, 131
- [61] C. LIU, P. FRAZIER, AND L. KUMAR. Comparative assessment of the measures of thematic classification accuracy. *Remote Sensing of Environment*, **107**[4]:606 – 616, 2007. 19, 20
- [62] D. LOWE. Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision*, **60**[2]:91–110, November 2004. 12, 16, 17, 31, 64
- [63] Y. LOYA. The coral reefs of eilat past, present and future: Three decades of coral community structure studies. In *Coral Health and Disease*, pages 1–34. Springer Berlin Heidelberg, 2004. 125, 131

REFERENCES

- [64] E. LUNDBLAD, D. WRIGHT, D. NAAR, B. DONAHUE, J. MILLER, E. LARKIN, AND R. RINEHART. Classifying deep water benthic habitats around tutuila, american samoa, 2004. 69
- [65] M. MARCOS, A. SHIELA, L. DAVID, E. PEAFLOP, V. TICZON, AND M. SORIANO. Automated benthic counting of living and non-living components in ngedarrak reef, palau via subsurface underwater video. *Environmental Monitoring and Assessment*, **145**:177–184, 2008. x, 11, 14, 17, 18, 24, 92, 93
- [66] M. MARCOS, M. SORIANO, AND C. SALOMA. Classification of coral reef images from underwater video using neural networks. *Optics Express*, **13**:8766–8771, 2005. 12
- [67] ZOLTAN-CSABA MARTON, DEJAN PANGERCIC, NICO BLODOW, AND MICHAEL BEETZ. Combined 2d-3d categorization and classification for multimodal perception systems. *Int. J. Rob. Res.*, **30**[11]:1378–1402, September 2011. 15
- [68] Y. MATSUSHITA, K. NISHINO, K. IKEUCHI, AND M. SAKAUCHI. Shadow elimination for robust video surveillance. *Motion and Video Computing, IEEE Workshop on*, page 15, 2002. 27
- [69] A. MEHTA, E. RIBEIRO, J. GILNER, AND R. WOESIK. Coral reef texture classification using support vector machines. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP), 07*, pages 302–310, 2007. 11, 14
- [70] S. MIKA, G. RATSCH, J. WESTON, B. SCHOLKOPF, AND K. MULLER. Fisher discriminant analysis with kernels. In *Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop.*, pages 41–48, 1999. 12, 74
- [71] E. NASSET. Conditional tau coefficient for assessment of producer’s accuracy of classified remotely sensed data. *ISPRS Journal of Photogrammetry and Remote Sensing*, **51**[2]:91 – 98, 1996. 20, 22
- [72] T. NICOSEVICI, N. GRACIAS, S. NEGAHDARIPOUR, AND R. GARCA. Efficient three-dimensional scene modeling and mosaicing. *J. Field Robotics*, **26**[10]:759–788, 2009. 64

REFERENCES

- [73] T. OJALA, M. PIETIKAINEN, AND D. HARWOOD. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, **29**[1]:51–59, January 1996. 11, 12, 17, 59
- [74] T. OJALA, M. PIETIKAINEN, AND T. MAAA. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, pages 971–987, 2002. 62
- [75] S. OUGIAROGLOU AND G. EVANGELIDIS. Fast and accurate k-nearest neighbor classification using prototype selection by clustering. In *Informatics (PCI), 2012 16th Panhellenic Conference on*, pages 168–173, 2012. 75
- [76] G. PADMAVATHI, M. MUTHUKUMAR, AND S. THAKUR. Kernel principal component analysis feature detection and classification for underwater images. In *Image and Signal Processing (CISP), 2010 3rd International Congress on*, **2**, pages 983–988, Oct. 2010. 12, 14
- [77] J. PAUL AND C. RAMESH. Invariant surface characteristics for 3d object recognition in range images. *Computer Vision, Graphics, and Image Processing*, **33**[1]:33–80, 1986. 15
- [78] N. PICAN, E. TRUCCO, M. ROSS, D. LANE, Y. PETILLOT, AND I. RUIZ. Texture analysis for seabed classification: co-occurrence matrices vs. self-organizing maps. In *OCEANS'98 IEEE Conference Proceedings*, **1**, pages 424–428 vol.1, Sep-1 Oct 1998. 10, 14
- [79] A. PIERRE, T. STEPHANE, AND W. CAMILLE. 3d fast intersection and distance computation (aabb tree). In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.3 edition, 2013. 66
- [80] O. PIZARRO, P. RIGBY, M. JOHNSON-ROBERSON, S. WILLIAMS, AND J. COLQUHOUN. Towards image-based marine habitat classification. In *OCEANS'08 MTS/IEEE Conference, Quebec*, Sept. 2008. x, 12, 14, 16, 24, 93, 94, 95
- [81] R. PORTER AND N. CANAGARAJAH. Robust rotation-invariant texture classification: wavelet, gabor filter and gmrf based schemes. *Vision, Image and Signal Processing, IEE Proceedings -*, **144**[3]:180–188, jun 1997. 57, 80

REFERENCES

- [82] D. POWERS. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. Technical Report SIE-07-001, School of Informatics and Engineering, Flinders University, Adelaide, Australia, 2007. 82
- [83] R. PRADOS, R. GARCIA, N. GRACIAS, J. ESCARTIN, AND L. NEUMANN. A novel blending technique for underwater gigamosaicing. *Oceanic Engineering, IEEE Journal of*, **37**[4]:626–644, 2012. 103, 131
- [84] E. REINHARD, M. ADHIKHMEN, B. GOOCH, AND P. SHIRLEY. Color transfer between images. *Computer Graphics and Applications, IEEE*, **21**[5]:34–41, 2001. 49, 50
- [85] C. ROTHER AND S. CARLSSON. Linear multi view reconstruction and camera recovery. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, **1**, pages 42 –50 vol.1, 2001. 19, 31
- [86] D. RUDERMAN AND W. BIALEK. Statistics of natural images: Scaling in the woods. In *Neural Information Processing Systems. NIPS*, pages 551–558, 1993. 75
- [87] GEORGIA SANDBACH, STEFANOS ZAFEIRIOU, MAJA PANTIC, AND DANIEL RUECKERT. Recognition of 3d facial expression dynamics. *Image and Vision Computing*, **30**[10]:762 – 773, 2012. 3D Facial Behaviour Analysis and Understanding. 15
- [88] B. SAREL AND M. IRANI. Separating transparent layers through layer information exchange. In *European Conference on Computer Vision ECCV 2004*, **3024** of *Lecture Notes in Computer Science*, pages 328–341. Springer Berlin / Heidelberg, 2004. 27
- [89] Y. SCHECHNER AND N. KARPEL. Attenuating natural flicker patterns. In *IEEE TECHNO-OCEAN, 04*, **3**, pages 1262 –1268, nov 2004. 5, 27
- [90] Y. SCHECHNER AND N. KARPEL. Recovering scenes by polarization analysis. In *OCEANS'04 IEEE Conference*, **3**, pages 1255 –1261 Vol.3, nov. 2004. 27
- [91] R. SCHETTINI AND S. CORCHS. Underwater image processing: State of the art of restoration and image enhancement methods. *EURASIP Journal on Advances in Signal Processing*, **2010**[1]:746052, 2010. 49

REFERENCES

- [92] A. SCHICK, M. FISCHER, AND R. STIEFELHAGEN. Measuring and evaluating the compactness of superpixels. In *21st International Conference on Pattern Recognition (ICPR), 2012*, pages 930–934, 2012. 54
- [93] T. SERRE, L. WOLF, AND T. POGGIO. Object recognition with features inspired by visual cortex. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, **2**, pages 994 – 1000 vol. 2, June 2005. 11
- [94] A. SHIHAVUDDIN, N. GRACIAS, AND R. GARCIA. Online sunflicker removal using dynamic texture prediction. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VIS-APP)*, pages 161–167. SciTePress, 2012. i, 5, 6, 28, 54
- [95] A. SHIHAVUDDIN, N. GRACIAS, R. GARCIA, J. ESCARTIN, AND R. PEDERSON. Automated classification and thematic mapping of bacterial mats in the north sea. In *OCEANS’13 MTS/IEEE Conference, Bergen*, June 10-13, 2013. i, 6, 128
- [96] A. SHIHAVUDDIN, N. GRACIAS, R. GARCIA, A. GLEASON, AND B. GINTERT. Image-based coral reef classification and thematic mapping. *Remote Sensing*, **5**[4]:1809–1841, 2013. i, 6, 12
- [97] N. SINHA, J. FRAHM, M. POLLEFEYS, AND Y. GENC. Feature tracking and matching in video using programmable graphics hardware. *Machine Vision and Applications*, **22**[1]:207–217, 2011. 65
- [98] L. SOH AND C. TSATSOU LIS. Texture analysis of sar sea ice imagery using gray level co-occurrence matrices. *Geoscience and Remote Sensing, IEEE Transactions on*, **37**[2]:780–795, 1999. 59, 60, 72, 80, 90
- [99] M. SORIANO, S. MARCOS, C. SALOMA, M. QUIBILAN, AND P. ALINO. Image classification of coral reef components from underwater color video. In *OCEANS, 2001. MTS/IEEE Conference and Exhibition*, **2**, pages 1008 –1013 vol.2, 2001. 11, 14
- [100] M. STOKES AND G. DEANE. Automated processing of coral reef benthic images. In *Limnology and Oceanography: Methods*, **7**, pages 157 –168, 2009. x, 12, 14, 17, 18, 24, 76, 80, 93

REFERENCES

- [101] M. STORY AND R. CONGALTON. Accuracy assessment - a user's perspective. *Photogrammetric Engineering and Remote Sensing*, **52**[3]:397–399, March 1986. 20
- [102] G. TOUSSAINT. Geometric proximity graphs for improving nearest neighbour methods in instance-based learning and data mining. *International Journal of Computational Geometry and Applications*, **15**[02]:101–150, 2005. 75
- [103] B. TRIGGS, F. McLAUCLAN, R. HARTLEY, AND W. FITZGIBBON. Bundle adjustment a modern synthesis. In BILL TRIGGS, ANDREW ZISSERMAN, AND RICHARD SZELISKI, editors, *Vision Algorithms: Theory and Practice*, **1883** of *Lecture Notes in Computer Science*, pages 298–372. Springer Berlin Heidelberg, 2000. 64
- [104] W. TURNER, S. SPECTOR, N. GARDINER, M. FLADELAND, E. STERLING, AND M. STEININGER. Remote sensing for biodiversity science and conservation. *Trends in Ecology Evolution*, **18**[6]:306 – 314, 2003. 1
- [105] Y. UKRAINITZ AND M. IRANI. Aligning sequences and actions by maximizing space-time correlations. In ALEX LEONARDIS, HORST BISCHOF, AND AXEL PINZ, editors, *European Conference on Computer Vision ECCV 2006*, **3953** of *Lecture Notes in Computer Science*, pages 538–550. Springer Berlin / Heidelberg, 2006. 27
- [106] J. VAN DE WEIJER, T. GEVERS, AND A. W M SMEULDERS. Robust photometric invariant features from the color tensor. *Image Processing, IEEE Transactions on*, **15**[1]:118–127, 2006. 64
- [107] J. VAN DE WEIJER AND C. SCHMID. Coloring local feature extraction. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part II, ECCV'06*, pages 334–348, Berlin, Heidelberg, 2006. Springer-Verlag. 63, 64, 80
- [108] M. VARMA AND A. ZISSERMAN. A statistical approach to texture classification from single images. *International Journal of Computer Vision. IJCV 05*, **62**[1-2]:61–81, April 2005. 13

REFERENCES

- [109] Y. WEISS. Deriving intrinsic images from image sequences. In *Computer Vision, ICCV 2001. Proceedings. Eighth IEEE International Conference on*, **2**, pages 68–75, 2001. 27
- [110] J. ZHANG, S. LAZEBNIK, AND C. SCHMID. Local features and kernels for classification of texture and object categories: a comprehensive study. *International Journal of Computer Vision. IJCV*, **73**:2007, 2007. 15, 95
- [111] X. ZHANG AND Y. JIA. A linear discriminant analysis framework based on random subspace for face recognition. *Pattern Recognition*, **40**[9]:2585–2591, 2007. 73
- [112] K. ZUIDERVELD. Contrast limited adaptive histogram equalization. pages 474–485. Academic Press Professional, Inc., San Diego, CA, USA, 1994. 52