Image processing algorithms for waterborne pathogen classification

Luo, Jiawen

2021

Luo, J. (2021). Image processing algorithms for waterborne pathogen classification.
Doctoral thesis, Nanyang Technological University, Singapore.
<https://hdl.handle.net/10356/154817>

<https://hdl.handle.net/10356/154817>

<https://doi.org/10.32657/10356/154817>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0
International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 19:22:23 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

IMAGE PROCESSING ALGORITHMS FOR WATERBORNE PATHOGEN CLASSIFICATION

LUO JIAWEN

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of Doctor of
Philosophy

2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

14 July 2021

Date

NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU

LUO Jiawen

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

13 July 2021

Date

NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU

Professor LU Yilong

Authorship Attribution Statement

This thesis contains material from 2 papers accepted at conferences, and from 3 paper(s) in the following peer-reviewed journals in which I am listed as an author.

- [1] J. Luo, and W. Ser, "Image Classification of Microorganisms Using Mutual Information Approach," Apr. 2018. in *IEEE International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018. (Abstract track)
- [2] J. Luo, A. Liu, P.H. Yap, B. Liedberg, and W. Ser, "An mRMR-SVM Approach for Opto-Fluidic Microorganism Classification," Jul. 2018. in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 666-669, IEEE, 2018.
- [3] J. Luo, W. Ser, A. Liu, P.H. Yap, B. Liedberg, and S. Rayatpisheh, "Waterborne Pathogen Classification with Optofluidic Images," *Biomedical Signal Processing and Control*. (Submitted)
- [4] J. Luo, W. Ser, A. Liu, P.H. Yap, B. Liedberg, and S. Rayatpisheh, "Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) for Microorganism Image Classification," *Multidimensional Systems and Signal Processing*. (Submitted)
- [5] J. Luo, W. Ser, A. Liu, P.H. Yap, B. Liedberg, and S. Rayatpisheh, "Microorganism Image Classification with Circle-Based Multi-Region Binarization and Mutual-Information-Based Feature Selection," *Biomedical Engineering Advances*, vol. 2, pp. 100020, 2021.

Chapter 4 is published as [1][2].

Chapter 4 and 7 are presented as [3].

Chapter 5 and 7 are presented as [4].

Chapter 6 and 7 are presented as [5].

The contributions of the authors of paper [1] are as follows:

- I (J. Luo) developed the algorithm, implemented the experiments, and prepared the manuscript.
- W. Ser provided the initial research direction and supervised the study.

The contributions of the authors of paper [2] are as follows:

- I (J. Luo) designed the algorithm by investigating the existing feature and proposing a new feature, implemented the experiments of the proposed and existing algorithms, and prepared the manuscript draft.
- A. Liu provided the optofluidic image dataset and assisted on the imaging technique explanation.
- P.H. Yap contributed with suggestions on image understanding.
- B. Liedberg made language revision of the manuscript.
- W. Ser supervised the study and manuscript writing.

The contributions of the authors of paper [3] are as follows:

- I (J. Luo) designed the algorithm, analyzed the feature rationale, implemented the experiments, interpreted the experimental results, and prepared the manuscript draft.
- W. Ser supervised the study and manuscript writing.
- A. Liu provided the optofluidic image dataset and assisted on the imaging technique explanation.
- P.H. Yap contributed with suggestions on the algorithm design.
- B. Liedberg revised the manuscript.
- S. Rayatpisheh contributed with suggestions on conventional microorganism detection methods.

The contributions of the authors of paper [4] are as follows:

- I (J. Luo) conceived the idea of Multi-Region Binarization, developed the algorithm, implemented the experiments on our dataset and the open-sourced dataset, and prepared the manuscript draft.
- W. Ser supervised the study and manuscript writing.

- A. Liu provided the optofluidic image dataset and assisted on the imaging technique explanation.
- P.H. Yap contributed with suggestions on image processing.
- B. Liedberg revised the manuscript.
- S. Rayatpisheh contributed with suggestions on open-sourced datasets.

The contributions of the authors of paper [5] are as follows:

- I (J. Luo) developed the algorithm, analyzed the data in our dataset and the open-sourced dataset, implemented the experiments, and prepared the manuscript draft.
- W. Ser supervised the study and manuscript writing.
- A. Liu provided the optofluidic image dataset and assisted on the imaging technique explanation.
- P.H. Yap contributed with suggestions on feature selection.
- B. Liedberg revised the manuscript.
- S. Rayatpisheh contributed with suggestions on open-sourced datasets.

01 January 2022

Date

NTU NTU NTU NTU NTU NTU NTU NTU

 NTU NTU NTU NTU NTU NTU NTU NTU
 NTU NTU NTU NTU NTU NTU NTU NTU
 NTU NTU NTU NTU NTU NTU NTU NTU

LUO Jiawen

Acknowledgements

I would like to express my sincere and deep gratitude to my supervisors, Associate Professor SER Wee and Professor LU Yilong, for their invaluable guidance, support, and encouragement. Associate Professor SER Wee provided me with the opportunity to start the research study and has been always approachable and patient to lead me on all aspects of the research like problem defining, data analyzing, method developing, and result presenting throughout my whole PhD study. After his retirement in June 2021, Professor LU Yilong kindly agreed to be my supervisor and gave me much advice and help on the PhD graduation processes including the thesis. The research journey with them has been an unforgettable experience guiding me in my future study and career.

I wish to thank the professors and colleagues in the Centre for Bio Devices and Signal Analysis (VALENS) Laboratory of Nanyang Technological University (NTU), especially Professor LIU Aiqun, Dr. LEI Lei, Dr. CHIN Lip Ket, and Dr. NGUYEN Kim Truc for providing the dataset and explaining the imaging procedures. I also wish to thank the Thesis Advisory Committee members, Associate Professor SER Wee, Professor LIU Aiqun and Assistant Professor LIN Guosheng, for all the suggestions and assistance during my PhD study. Many thanks to the faculties and staff in the School of Electrical and Electronic Engineering (EEE), NTU for the professional courses, administrative and technical support, and friendly working environment.

The financial support of the NTU Research Scholarship from the NTU is greatly appreciated.

I am grateful to my family, friends, and boyfriend for the love, companion, and encouragement. Last but not the least, my gratefulness goes to my parents for creating a nice environment for me to grow up and providing constant financial and emotional support to complete my study. Their unconditional love is the most precious treasure in my life.

Contents

Statement of Originality	i
Supervisor Declaration Statement	ii
Authorship Attribution Statement	iii
Acknowledgements	vi
Contents	vii
Summary	xii
List of Abbreviations	xiii
1 Introduction	1
1.1 Motivation	1
1.2 Objectives	3
1.3 Organization of the Thesis	3
2 Literature Review	7
2.1 Optofluidic Systems for Microorganism Image Capturing	7
2.2 Machine Learning Algorithms for Image Classification	8
2.2.1 Handcrafted-Feature-Based Approach	9
2.2.2 Deep-Learning-Based Approach	15
2.2.3 Reason to Adopt Handcrafted-Feature-Based Approach	17
3 Datasets Used	19
3.1 Optofluidic Images of Waterborne Pathogenic Protozoa	19
3.2 Transmission Electron Microscope (TEM) Images of Virions	22

4	Entropy of Histogram of HOG for Waterborne Pathogen Classification	25
4.1	Introduction	25
4.2	Investigation on Various Features	25
4.2.1	Feature Category	26
4.2.2	Boxplot Check	29
4.2.3	Analysis on Features of Different Categories	31
	Morphological Features	32
	Optical Features	33
	Textual Features	34
	Other Features	34
4.2.4	Feature Investigation Summary	35
4.3	Proposed Feature: Entropy of Histogram of Histogram of Oriented Gradients (EHHOG)	36
4.3.1	Histogram of Oriented Gradients (HOG)	36
	Partial Derivatives	36
	Image Gradient	37
	Orientation Discretization	38
	Weighted Voting Values	38
4.3.2	Entropy of Histogram of Histogram of Oriented Gradients (EHHOG)	39
	Calculation Procedures	39
	Boxplot Check	40
	Visualization of HOG	42
	Visualization of Histogram of HOG	44
4.4	Classification Results and Discussions	49
4.4.1	Using Different Classifiers	49
	Support Vector Machine and Bagging Decision Trees	50
	Artificial Neural Networks with Different Configurations	51
4.4.2	Using Existing Microorganism Classification Methods on Our Dataset	56
4.5	Summary of Proposed EHHOG Algorithm	59
5	Multi-Region Binarization to Improve Waterborne Pathogen Classification	60
5.1	Introduction	60

5.2	Proposed Multi-Region Binarization Method	60
5.2.1	Multi-Region Binarization	61
5.2.2	Comparing Binarized Images with Otsu and Multi-Region Binariza- tion Method	62
5.3	Feature Combination	63
5.3.1	Features from Binarized Images	63
5.3.2	Features from Greyscale Images	64
5.3.3	Feature Combination Used in Classification Experiments	67
5.4	Classification Results and Discussions	68
5.4.1	Different Feature Combinations	68
5.4.2	Other Local Thresholding Binarization Methods	72
	Binarized Image Comparison	72
	Classification Result Comparison	82
5.5	Summary of Proposed BiGHAM Algorithm	83

6 Mutual-Information-Based Feature Selection to Improve Waterborne Pathogen

Classification	85
6.1 Introduction	85
6.2 Mutual-Information-Based Feature Selection	86
6.2.1 Feature Ranking Criterion	86
6.2.2 Ways to Apply Feature Selection	87
Feature Ranking on All Features	87
Feature Ranking on Each Morphological Feature Separately	87
6.3 Rectangle-Based and Circle-Based Multi-Region Binarization	88
6.3.1 Comparison of Binarized Images	89
6.3.2 Comparison of Boxplot Check Results	93
6.4 Classification Results and Discussions	94
6.4.1 Experiments on Our Original Dataset	94
mRMR Ranking on All Features	94
Feature Combination <i>G1B1Bα</i> and <i>G1B1ARSOCO</i> with with Rectangle- Based and Circle-Based Regions	97
6.4.2 Experiments on Our Dataset with Different Sample Numbers	98

Proposed EHHOG Method	98
Proposed Feature Combination <i>G1B1Bα</i>	99
Proposed Feature Combination <i>G1B1ARSOCO</i>	101
Existing Microorganism Image Classification Algorithms	106
6.5 Summary of Proposed iBiGHAM Algorithm	109
7 Adaptation and Application of Proposed Algorithms on Open-Sourced Data	112
7.1 Introduction and Pre-Processing on Open-Sourced Dataset BCO-DMO 670706	112
7.2 Adjustment on Data Augmentation and Binarization Procedures	113
7.2.1 Padding Methods for Data Augmentation	113
7.2.2 Binarization Settings	113
7.2.3 Image Illustration	114
7.2.4 Boxplot Check Results	119
7.3 Classification Results and Discussions	126
7.3.1 Proposed Algorithms	126
7.3.2 Existing Microorganism Image Classification Algorithms	136
7.4 Summary	139
8 Conclusions and Future Outlook	141
8.1 Conclusions	141
8.2 Recommendations for Future Work	143
8.2.1 Deep Learning Approaches	144
Fine-Tuning Settings for Deep Learning on Our Dataset	146
Fine-Tuning Settings for Deep Learning on Open-sourced Dataset	
BCO-DMO 670706	148
Lightweight Deep Learning Models	152
Summary	153
8.2.2 Water Quality Monitoring and Other Applications	153
Author's Publications	156
Bibliography	157
A Appendix A - Boxplot Check on Features in Kumar's Method	170

B	Appendix B - Boxplot Check on Features in Bueno's Method	174
C	Appendix C - Difference between HOG Entropy and EHHOG	224
D	Appendix D - One or Two Repetitions of 10-Fold Cross-Validation	227
E	Appendix E - Experiment Results of Deep Learning	229
E.1	Experiment Results of Pedraza's method on Our Dataset	232
E.2	Experiment Results of Pedraza's method on Open-Sourced Dataset BCO- DMO 670706	240
E.3	Experiment Results of MobileNetV2 on Our Dataset	250
F	Appendix F - Calculation Method of Aspect Ratio, Solidity, and Compactness	251
G	Appendix G - Computational Complexity Calculation Method and Example	254
G.1	Introduction	254
G.2	Calculation Example	255
G.3	Open-Sourced Tools	257
H	Appendix H - Examples of Rotated Images and Corresponding Features	259
H.1	Image Example of Our Dataset	260
H.2	Image Example of Open-Sourced Dataset BCO-DMO 670706	263

Summary

Waterborne pathogens are microorganisms in the water that can cause infections, diseases and even deaths. Classification of such microorganisms is essential for water quality monitoring which is important for human health and safety. The conventional biochemical approaches are standard, accurate but have high requirements on monetary, time, and human resources, and usually leave chemical pollutions. With the development of machine learning techniques, algorithms were established for cheaper and simpler microorganism image classification.

This PhD study has investigated and developed portable and accurate classification algorithms that are more feasible for portable, real-time applications based on images captured by an optofluidic system, which is smaller and cheaper than microscopes. Most existing microorganism classification algorithms are designed for microscopic images so do not work well on our optofluidic images. Besides, the existing algorithms are mainly focused on classification accuracy but not on the computational cost. Thus, this PhD study aims to develop image processing algorithms for waterborne pathogen classification that have the potential for portable, real-time water quality monitoring applications.

This PhD study has made three contributions. The first contribution is the proposed EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) method, combining some existing morphological features extracted from binarized images and the proposed feature EHHOG extracted from greyscale images, for waterborne pathogen classification. The second contribution is proposing the Multi-Region Binarization method to capture the information missed by the existing binarization methods and developing the Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) method that improved the classification accuracy from the EHHOG method. The third contribution is the proposed improved-BiGHAM (iBiGHAM) method with a new circle-based region partitioning for Multi-Region Binarization and a new way to apply the mutual-information-based feature selection. The iBiGHAM method further increased the classification accuracy with simpler trial procedures than the BiGHAM method. It could achieve similar or higher accuracy than the existing algorithms including the deep-learning-based approach, while the computational complexity was much lower. Moreover, these proposed algorithms were adapted and applied to images in an open-sourced dataset, also achieving better classification performance than the existing algorithms.

List of Abbreviations

ANN	Artificial Neural Network
AR	Aspect Ratio , ratio of major axis length and the minor axis length
BDT	Bagging Decision Trees
BiGHAM	Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization
CNN	Convolutional Neural Network
CO	COmpactness
CPU	Central Processing Unit
CV	Cross-Validation
DT	Decision Tree
EHOG	Entropy of Histogram of HOG
FC	Fully-Connected
FLOP	FLloating-point OPeration
FE	Feature Extraction
FS	Feature Selection
GLCM	Grey Level Co-occurrence Matrix
GPU	Graphics Processing Unit
HOG	Histogram of Oriented Gradients
iBiGHAM	improved Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization
IEN	Intensity ENtropy , entropy of intensities
IFS	Incremental Feature Selection
k-NN	k- Nearest Neighbour
LR	Learning Rate
mRMR	minimal-Redundancy-Maximal-Relevance
NMF	Non-negative Matrix Factorization
NTU	Nanyang Technological University
PC	Principal Component

PCA	P rincipal C omponent A nalysis
PhD	D octor of P hilosophy
RBF	R adial B asis F unction
ReLU	R ectified L inear U nit
RLN	R un L ength N on-uniformity
SGD	S tochastic G radient D escent
SO	S OLIDity
SVM	S upport V ector M achine
TEM	T ransmission E lectron M icroscope
WHO	W orld H ealth O rganization

Chapter 1

Introduction

1.1 Motivation

Microorganisms exist prevalently in foods, water, soil and the environment. Among them, pathogens are those that can cause infections, diseases and even deaths. Hence, microorganism analysis is crucial for human health and well-being. In the water supplies, the waterborne pathogens including viruses, bacteria, protozoa, and helminths form an important hazard source, especially in rural and undeveloped areas [1]–[5]. (Table 1.1) Unlike bacteria, viruses, and helminths, protozoa have high infectivity and high resistance to chemical disinfection like chlorine [1]. Protozoan infection causes more than 58 million cases of childhood protozoal diarrhoea per year, and the direct costs are estimated to be US\$ 150 million [2]. Infectious diarrhoea is a major cause of death in children under 5 years old in Africa [3].

Giardia and *Cryptosporidium* are two common protozoa with important animal sources, related to health risks such as diarrhoea, abdominal cramps, nausea, vomiting, fever, and even death in severe cases, and can keep infective more than one week time in the water [1],[6]. They were included in the “Neglected disease initiative” by WHO due to their link with poverty and significant disease burden in developing countries [2]. *Giardia* infects approximately 200 million people worldwide; *Cryptosporidium* is among the leading causes of moderate to severe diarrhoea in children under 2 years [6]. The largest *Cryptosporidium* outbreak in Milwaukee, USA in 1993 made about 1.5 million people exposed. Among them, 403 thousand people became ill and 104 died. The total cost of outbreak-associated illness was US\$ 96.2 million [7]. In summary, *Giardia* and *Cryptosporidium* have high health significance and are a major concern for water supply safety [1],[4],[8]–[10]. Thus, detection of such microorganisms is a crucial step in water quality monitoring.

TABLE 1.1: Waterborne pathogen categories and sizes

Summarized from WHO's guidelines [1] and the book chapters [11]–[14]. $1 \text{ mm} = 1 \times 10^{-3} \text{ m}$. $1 \mu\text{m} = 1 \times 10^{-6} \text{ m}$. $1 \text{ nm} = 1 \times 10^{-9} \text{ m}$.

Category	Typical size
Virus	10 to 300 nm
Bacterium	0.1 to 10 μm
Protozoan	5 to 50 μm
Helminth	Longer than 1 mm, even up to 10 m Helminths are not microorganisms strictly speaking, but studied in microbiology because many infections are caused by eggs and larvae which are very small (e.g., 50 μm).

The conventional methods based on biochemical approaches remain the golden standard for microorganism detection. However, they are time-consuming labour-intensive and may cause chemical pollutions. Several days are often required for a complete process and tasks like culture medium preparation, inoculation, culturing, and biochemical assays must be done by trained professionals in qualified laboratories [9],[15]–[17].

Besides the conventional methods, machine-learning-based image classification algorithms are also developed for microorganism detection recently [18],[19]. Microorganisms can be identified under microscopes by manual observations of a microbiological expert according to the morphology information (shape of the microorganism), indicating that different types of microorganisms have different appearances on the image. Thus, with the development of machine learning, image processing and classification algorithms have been applied for microorganism classification for the past few decades. These machine-learning-based algorithms were faster and more automatic than the conventional methods, but most of them were designed for images captured by microscopes, which were not always suitable for the optofluidic images in our dataset.

As the imaging technique, the emerging optofluidic systems bring new possibilities of cheaper, faster and easier microorganism classification. An optofluidic system is a small, on-chip device combining optics and microfluidics that can be used to imaging waterborne microorganisms [20]–[22]. Establishing waterborne pathogen classification methods with such

low-cost optofluidic imaging techniques can promote portable, real-time water quality monitoring and benefit human health and well-being, especially in rural and undeveloped areas. In optofluidic systems, the sample is pumped through microfluidic channels on a chip and the microorganisms/particles are controlled by the optical force for sensing and imaging. In this way, single-cell images suitable for machine-learning-based classification algorithms can be easily captured. No culturing or microscope specimen preparation is required before the imaging, thus a lot of time and effort is saved.

1.2 Objectives

Microorganism classification is crucial for water quality monitoring but the conventional methods are expensive, complicated, require trained experts, and take much time. Thus, this PhD study aims to develop image processing algorithms that can automatically classify waterborne pathogens using optofluidic images and potential for portable, real-time water quality monitoring applications to ensure the safety of drinking water sources.

An optofluidic microorganism imaging system is smaller, faster, and cheaper than conventional microscopes, so more feasible for portable, real-time applications (details in Section 2.1). The optofluidic images in our dataset show the scattering patterns outside the microorganism cells/ particles, while many microscopic images show the texture inside the cell (details in Section 3.1). The existing microorganism classification algorithms were not suitable for our optofluidic images or had high computational requirements which made it difficult for portable, real-time applications (experiment results presented and discussed later). Thus, new algorithms are expected to distinguish different classes of optofluidic images at a low computational cost.

1.3 Organization of the Thesis

This thesis is mainly organized into 8 chapters plus references and 8 appendices.

Chapter 1 introduces the motivation and objective of this PhD study. Waterborne pathogens are microorganisms in the water that can cause human diseases, so microorganism classification is crucial for water quality monitoring. The conventional biochemical methods are

expensive, complicated, and slow. Thus, this PhD study aims to develop microorganism classification algorithms based on optofluidic images of waterborne pathogens that are potential for portable, real-time water quality monitoring applications.

Chapter 2 summarizes the literature study. For the microorganism imaging technique, optofluidic systems and conventional microscopes are compared, showing that optofluidic imaging is more feasible for portable, real-time applications. Machine learning algorithms for image classification and examples of applications on microorganism images are enumerated. The reason for adopting the handcrafted-feature-based approach is discussed upon the comparison between the handcrafted-feature-based approach and the deep-learning-based (DL-based) approach.

Chapter 3 introduces the datasets used in this PhD study. The optofluidic images in our dataset containing 3 classes, *Giardia*, *Cryptosporidium* (2 common waterborne pathogens) and others (non-hazardous particles in water), were used to design the proposed algorithms. An open-sourced dataset BCO-DMO 670706 [23] with 2 classes of virion images captured by a transmission electron microscope (TEM) was used to verify the proposed algorithms.

Chapter 4 presents the proposed EHHOG algorithm. Starting with categorizing and investigating over 1000 features used in the existing microorganism classification algorithms, boxplots showed that most of these features used for microscopic images were not suitable to distinguish the 3 classes of our optofluidic images, so only some of the morphological features extracted from the binarized images were chosen. To capture the information missed by the binarization process, a new feature based on the histogram of oriented gradients (HOG), the entropy of histogram of HOG (EHHOG), was proposed and proved to be useful for our classification task no matter in boxplots or classification experiments. The derivation steps and rationale visualizations of the feature EHHOG were displayed. This proposed algorithm combining the chosen morphological features and EHHOG was named EHHOG algorithm. The classification results and computational complexities of the proposed EHHOG algorithm and some existing microorganism classification algorithms were compared.

Chapter 5 presents the proposed BiGHAM algorithm. Since the conventional global-thresholding binarization method, Otsu's method [24], could only segment out the inner part of the scattering patterns, a new local-thresholding binarization method, Multi-Region Binarization, was proposed to capture the outer parts of the patterns on our optofluidic images. With the new binarization method, more features were generated, and a feature combination

$G1B1B\alpha$ was used, where $G1$ refers to the features from greyscale images, $B1$ refers to the features from Otsu binarized images, and $B\alpha$ refers to the features from Multi-Region binarized images. This proposed algorithm was named Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) algorithm. Classification experiments showed the accuracy increase from EHHOG algorithm to BiGHAM algorithm. Comparisons to some existing local-thresholding binarization methods were also presented and discussed.

Chapter 6 presents the proposed iBiGHAM algorithm. In the previous BiGHAM algorithm, the feature selection was made by experience and trials. In this improved-BiGHAM (iBiGHAM) algorithm, a mutual-information-based feature ranking criterion, minimal-Redundancy-Maximal-Relevance (mRMR) [25], was applied to select the feature combination $G1B1ARSOCO$ at one time. Instead of the conventional incremental feature selection (IFS) that required at least 10 times of training and testing, a new way to apply the mRMR criterion was proposed for our features that only need 1 time of training and testing and achieved higher classification accuracy. Besides, the rectangle-based Multi-Region Binarization used in BiGHAM algorithm was replaced by the circle-based Multi-Region Binarization. The classification accuracies increasing from EHHOG algorithm to BiGHAM algorithm to iBiGHAM algorithm for the original dataset and augmented datasets with 30, 70, 100, 300, 700, 1000 images per class was presented and discussed.

Chapter 7 shows the adaptations and classification results made for the proposed algorithms to be applied on the open-sourced dataset BCO-DMO 670706 [23]. With some simple adjustment, the proposed algorithms could successfully classify the 2 classes of TEM images in this open-sourced dataset, indicating that the proposed algorithms also had the potential to be extended to other applications with microscopic images. Classification experiments of the proposed algorithms and some existing microorganism classification algorithms on the original dataset and augmented datasets were compared.

Chapter 8 concludes the thesis and discusses the directions for future researches. Although deep learning (DL) is not used in the proposed algorithms for it has high requirements of computational power and dataset size and thus are not very suitable for portable applications, it is worth studying as an emerging machine learning approach for other image classification applications. Despite the convenience of not handcrafting features, the settings for fine-tuning when applying a DL model to a new dataset require manual determination by trials. Some preliminary experimental results were presented in this chapter, leaving the question of

how to determine the DL settings in a more automatic way for future researches. Another direction for future work is to establish some water quality monitoring devices/systems based on the current algorithms. Possible studies include investigating more types of microorganisms, optimizing the proposed algorithms, and embedding the algorithms on hardware.

Chapter 2

Literature Review

2.1 Optofluidic Systems for Microorganism Image Capturing

Optofluidics is an emerging field for the past 15 years combining optics and microfluidics synergistically. The differences comparing optofluidic imaging with microscopic imaging are listed in Table 2.1.

The functionalities of optofluidic systems are integrated on a chip with a size similar to a coin [26],[27] so they provide a small and cheap way to capture microorganism images instead of the conventional microscopes [21]. Besides the small instrumental size and low cost, optofluidic imaging is also faster than microscopes because the images are taken as the fluid sample flows through the microfluidic channels. Although state-of-the-art optofluidic devices are still commonly developed for laboratory uses and required to be operated by experienced professionals like the microscopes, the integration, compactness, and miniaturization of optofluidic chips make them promising for on-site analysis (e.g., portable, real-time water quality monitoring near the water reservoirs instead of collecting and sending the samples to distant laboratories) [22],[28]. For typical microscopic images, one image often contains many microorganisms, so to obtain single-cell images, tedious manual pre-processing or complicated segmentation algorithms should be applied [29]–[31]. In an optofluidic system, optical forces can be applied to trap microorganism cells/ particles, allowing convenient single-cell image capturing [21],[22],[32]. Optofluidic imaging also has some disadvantages compared to microscopic imaging like the limitations on sample types and image resolution. Optofluidic systems are not as powerful as electron microscopes that can achieve resolution at about 0.1 nm [33] to capture images of viruses (sizes shown in Table 1.1), but for analyzing protozoa and some bacteria, optofluidic imaging is good enough as a low-cost, efficient and

useful imaging technique [20],[27],[34].

TABLE 2.1: Comparison of microscopic imaging and optofluidic imaging for microorganisms

$$1\ \mu\text{m} = 1 \times 10^{-6}\ \text{m}. \quad 1\ \text{nm} = 1 \times 10^{-9}\ \text{m}.$$

Microscopic imaging	Optofluidic imaging
Bulky machine	Small, compact, miniaturized, on-chip system
Expensive	Relatively cheap
Static, slow	Dynamic, fast
Requiring experienced professionals to operate	Relatively simple and automatic
Samples need to be collected and sent to a qualified laboratory	Feasible for on-site analysis
Tedious to obtain single-cell images	Easier to obtain single-cell images by employing optical forces with microfluidics to trap and manipulate the particles
Can analyse various samples	Only applicable to fluidic samples (There are pure fluid-based, solid-liquid-hybrid and colloid-base optofluidic systems developed, but the sample types are still limited compared to microscopes.)
High-resolution images (many electron microscopes have resolutions about 1 nm or even 0.1 nm)	Difficult to achieve image resolution as high as high-resolution microscopes (typical resolutions are at the level of 1 μm or 100 nm)

2.2 Machine Learning Algorithms for Image Classification

In machine learning, the most common form is supervised learning, which builds a model and makes predictions about new samples based on known samples with targets (labels) provided

[35],[36]. For image classification, the samples are the images, and the labels are the classes (types) of the images. Since our dataset with optofluidic images has clear labels, supervised machine learning is focused on in this PhD study.

2.2.1 Handcrafted-Feature-Based Approach

The traditional machine learning algorithms rely on handcrafted features to make predictions [35],[37]–[39]. The typical workflows of these handcrafted-feature-based approaches for image classification are summarized in Figures 2.1, 2.2.

In the training stage, pre-processing (sometimes regarded as a part of feature extraction), feature extraction, and feature selection (optional) of the training image generate the handcrafted feature set for classification. The classifier takes these features as the input and predicts the class for each image. Then the classifier will be adjusted iteratively (i.e., training the classifier) based on the difference between the predicted class and the real class. After completing the procedures with all the training images, the trained classifier will be used in the testing.

In the testing stage, the same pre-processing and feature extraction as in the training stage are implemented on the testing images. If feature selection is used in the training, then only the selected features are to be extracted for the testing images. Finally, the trained classifier will generate the predicted class from the features of the testing images. The evaluation of the algorithm performance is based on the predicted classes and the ground truth labels (real classes) of the testing images. The most frequently used evaluation metric is the classification accuracy [19],[38], calculated as the ratio of the number of testing images that were correctly predicted and the total number of the testing images.

FIGURE 2.1: Demonstration of the training stage of the handcrafted-feature-based approaches for image classification

FIGURE 2.2: Demonstration of the testing stage of the handcrafted-feature-based approaches for image classification

For microorganism image classification, the features used in existing algorithms can be categorized into 4 types: morphological, optical, textural, and other features (details presented and discussed in Section 4.2). Among them, the morphological features describing the object shape are most commonly used [19]. The frequently used classifiers include support vector machines (SVM), decision trees (DT)/ bagging decision trees (BDT), and artificial neural networks (ANN) [19],[37],[39],[40]. Among them, the most popular classifier is SVM [19],[40].

TABLE 2.2: Existing microorganism image classification algorithms with handcrafted-feature-based approaches

Year	Paper	Feature	Classifier	Data
------	-------	---------	------------	------

1994	M.-P. Dubuisson, et al. [41]	2 morphological features: (1) Covariance matrix eigenvalue ratio (2) Shape factor	Linear decision boundary: the ratio of 2 features < 30 and the ratio of 2 features ≥ 30 represent the 2 predicted labels	Multiple-cell microscopic images of 2 types of bacteria
2008	C. Chen, X. Li [42]	4 morphological features (area, perimeter, major axis length, minor axis length) and 7 other features (Hu moments) + principal component analysis (PCA)	Artificial neural network (ANN)	Microscopic images of 4 types of bacteria
2010	S. Kumar, G.S. Mittal [43]	2 morphological features, 3 optical feature, and 4 textural features (investigated in Appendix A)	ANN	Single-cell microscopic images of 5 types of bacteria
2010	P.S. Hiremath, P. Bannigidad [44]	5 morphological features (shape factor, compactness, eccentricity, tortuosity, aspect ratio)	3σ classifier, k-nearest neighbours (k-NN) classifier, ANN, fuzzy classifier	Multiple-cell microscopic images showing 3 types of bacterial cell growth phases

2012	P.S. Hiremath, et al. [45]	7 morphological features (elongation, compactness, eccentricity, tortuosity, aspect ratio, 1/solidity, ratio of convex perimeter and object perimeter)	3σ classifier: calculating the means and standard deviations of the 2 classes (features of the training images), then predicting the class of the testing image by whether the feature lies in the range of (mean $\pm$ standard deviation)	Scanning electronic microscopic (SEM) images of 2 types of bacteria (some are single-cell)
2013	C. Li, et al. [46]	Morphological features	Support vector machine (SVM)	EMDS [40]: Microscopic images of different types of environmental microorganisms (EMs) (some are single-cell)
2013	C. Li, et al. [29]	Morphological features	SVM	EMDS [40] (some are single-cell)
2014	N.A. Mohamad, et al. [47]	2 morphological features (diameter, shape)	Naive Bayes	Multiple-cell microscopic images of 3 types of bacteria

2014	C. Yang, et al. [30]	9 morphological features	SVM (With high-dimensional features and small number of image samples, SVM is better than other classifiers like k-NN and Naive Bayes [48])	EMDS [40] (some are single-cell)
2017	G. Bueno, et al. [49]	1460 feature including morphological, optical, textural, and other features + unsupervised feature selection (FS) based on feature similarity	Bagging decision trees (BDT)	Single-cell microscopic images of 80 types of diatoms
2021	A. Dhindsa, et al. [50]	24 morphological features + unsupervised FS (principal component analysis (PCA) not useful)	ANN, k-NN, quadratic discriminant analysis (QDA), logistic regression (LR), SVM (the best classifier: SVM)	Multiple-cell microscopic images of 10 types of microorganisms

The handcrafted features, classifiers, and image data used in existing microorganism classification algorithms are summarized in Table 2.2. They were all designed based on microscopic images that usually showed the texture inside the microorganism cell, so besides morphological features, many existing algorithms relied on textural features. Our dataset of optofluidic images do not have the texture inside the cell, but only scattering patterns outside it, so the existing features should be investigated and a new feature is proposed to classify our optofluidic images (Chapter 4). The existing algorithms by S. Kumar, G.S. Mittal [43] and G. Bueno, et al. [49] established based on single-cell images are applied on our dataset to be compared with the proposed algorithms.

Another difference between the microscopic images and optofluidic images is that the microscopic images usually contain many cells on one image. Sometimes, the existing algorithms used single-cell image datasets obtained from the original multiple-cell images by manual cropping [43],[49]. In other cases, the image datasets had multiple-cell images, so complicated segmentation techniques should be included in the classification algorithms. Image segmentation is an important pre-processing technique that separates the images into the background and different objects [51] and is very useful to improve the performance of image classification [52],[53]. Image segmentation for multiple objects often requires complicated algorithms costing a long processing time [54]–[56]. When there is only one target object in the images (i.e., single-cell images for microorganism classification), image segmentation can be simplified to image binarization as there are only 2 parts, background and foreground (object), to be separated. As discussed in Section 2.1, optofluidic systems are more advantageous to capturing single-cell images than microscopes, so the proposed algorithms can adopt simpler binarization techniques to ensure a low computational cost. In the PhD study, an image binarization method based on Otsu’s thresholding [24] is proposed to help classify waterborne pathogens with our optofluidic images (Chapter 5).

When the number of features increases, the classification accuracy may not increase and even drop, especially with a small number of samples. Facing this curse of dimensionality problem, proper dimensionality reduction is helpful to improve the performance of the algorithms. Dimensionality reduction can be done by feature transformation approach or feature selection approach. Feature transformation like principal component analysis (PCA) and non-negative matrix factorization (NMF) can be seemed to be a part of feature extraction because it generates new features. They are not adopted as discussed in Section 4.2.3. Feature selection is to select a subset of the original feature set that reduces the computational cost, provides a better understanding of the data, and improves the classification accuracy [57]–[60]. Based on the fact that our dataset has clear labels for each image, the mutual-information-based (MI-based) feature ranking criterion, minimal-Redundancy-Maximal-Relevance (mRMR) [25], was applied for feature selection. Unlike the usual way of using incremental feature selection (IFS) to apply mRMR [61]–[64], a new way is proposed to select the features to classify our optofluidic images (Chapter 6).

2.2.2 Deep-Learning-Based Approach

During the past several decades, deep learning (DL) has become a popular research topic in machine learning. DL is defined as models containing multiple information processing layers of non-linear operations [36],[65]–[67], for example, deep artificial neural networks (ANNs). There is no conclusive answer to when a network is considered deep, but usually, a model with at least 4 non-linear layers is regarded as a DL model [65],[67]. With the improvement of the computational power and dataset richness, the number of research papers on DL-based approaches has been increasing rapidly [68].

When DL-based approaches used for image classification, one single DL model can perform as the combination of a feature extractor and a classifier. It takes the image itself as the input and predicts the class label as the output, thus specific feature investigating and hand-crafting are no longer needed. Starting from the LeNet [69],[70] in 1998, the convolutional neural network (CNN) has become the most popular DL-based approach for image classification. The structures of a shallow artificial neural network (ANN) and a DL-based CNN, LeNet-5, are illustrated in Figure 2.3 and Figure 2.4 below. In Figure 2.3, each circle represents one neuron (unit), and the neurons in the hidden layers (layers except the input and output layers) are called hidden neurons. Every neuron (except the input) performs the non-linear operation (denoted as f), called the activation function. The most frequently applied activation function is rectified linear unit (ReLU) [71] at present [36],[67]. As for the DL model CNN (Figure 2.4), besides going deeper (more layers), 2-D convolutions on the image were added before the non-linear operations.

FIGURE 2.3: Illustration of the structure of the artificial neural network (ANN)

Image source: [Paper "Deep learning" \[36\]](#)

FIGURE 2.4: Illustration of the structure of LeNet-5 [70]

Image source: [Paper "Gradient-based learning applied to document recognition" \[70\]](#)

For microorganism image classification, the common way to apply deep-learning-based (DL-based) approaches is to use a DL model pre-trained on some large image datasets instead of training the model from the beginning due to the limitations on the number of microorganism images and computing resources [19]. Popular convolutional neural network (CNN) models include AlexNet [72] applied on microscopic diatom images by A. Pedraza, et al. [73] in 2017, VGG-16 [74] applied on microscopic environmental images by S. Kosov, et al. [40], and Inception-v3 [75],[76] applied on microscopic environmental images by C.-M. Liang, et al. [77]. Considering our aim of developing algorithms for future portable, real-time applications, the simplest (in terms of computational complexity) method [73] is used for comparison

in the experiments.

2.2.3 Reason to Adopt Handcrafted-Feature-Based Approach

Deep learning (DL) enables image classification without the requirement of handcrafting features, but it is not to conclude that all handcrafted-feature-based approaches should be replaced by DL-based approaches. Both approaches have their advantages and disadvantages, so should be chosen according to specific application scenarios [78].

TABLE 2.3: Comparison between handcrafted-feature-based and deep-learning-based (DL-based) machine learning approaches

Handcrafted-feature-based approaches	DL-based approaches
Rule-based	Data-driven
Easy to train	Training requires powerful computing resources and big data
Low computational complexity	High computational complexity
Need to extract and select features for specific tasks	Can be applied to different problems by transfer learning

FIGURE 2.5: Demonstration of handcrafted-feature-based and deep-learning-based (DL-based) machine learning approaches

Image source: Paper "Deep learning for smart manufacturing: Methods and applications" [79]

The differences between handcrafted-feature-based and deep-learning-based (DL-based) approaches are shown in Table 2.3 and Figure 2.5. Comparing to the handcrafted-feature-based approaches, DL-based approaches rely more on the dataset (DL requires a large number of samples and high-quality data), so described as data-driven or data-hungry approaches [79],[80]. Besides the high requirements on data, DL also requires powerful computing resources (e.g., graphics processing unit (GPU)) and a long time for training, especially when the DL models are becoming deeper and complicated nowadays, while handcrafted-feature-based models can often be trained on an ordinary workstation or even a laptop. As a result of building complicated models, the computational complexity of DL-based approaches is also much higher than handcrafted-feature-based approaches, meaning that DL-based models cost much more time to test one sample if other conditions are unchanged. DL saves the efforts on manual feature extraction and selection, but not fully automatic because tedious fine-tuning and algorithm adaptations are also necessary for many applications on new problems. The success of DL-based approaches is highly dependent on complicated models, plentiful computing resources and large datasets [78],[80]. Thus, based on the aim of developing algorithms for portable, real-time applications, the cheaper (in terms of the device required for model training and testing), simpler (in terms of computational complexity) and faster handcrafted-feature-based approaches are adopted in this PhD study.

Chapter 3

Datasets Used

3.1 Optofluidic Images of Waterborne Pathogenic Protozoa

As discussed in Chapter 2, an optofluidic system is smaller, simpler, and cheaper than the conventional microscopes to capture microorganism images, so more suitable for portable, real-time applications. In the optofluidic system that captured the images of our dataset, the water sample flows through the microfluidic channel on a chip, and the microorganisms/particles in it will interrupt the light from the top, then the image of the particle is taken from the bottom as shown in Figure 3.1.

FIGURE 3.1: Illustration of the optofluidic system capturing the images of our dataset

Reprinted with permission from [32].

Our dataset contains optofluidic images of the three classes listed below. *Giardia* and *Cryptosporidium* are two common waterborne pathogenic protozoa that have high health significance and have become a major concern in various water supplies around the world

[1],[3]–[5],[7],[8],[10]. Others refer to the non-hazardous particles in the water except *Giardia* and *Cryptosporidium*. The ground truth labels of class G and class C were from the sample provider. The class O images came from the tap water in Singapore. There is no *Giardia* and *Cryptosporidium* in the tap water according to the Public Utilities Board (PUB), Singapore’s National Water Agency [81].

- 1) *Giardia* (called class G later)
- 2) *Cryptosporidium* (called class C later)
- 3) Other particles (called class O later)

FIGURE 3.2: Examples of images in our dataset captured by the optofluidic system

Numbers of image samples for class G, C, and O are 147, 117, and 148 respectively.

The images in our dataset were single-cell grayscale images showing the scattering patterns manually cropped to 488*648 pixels. This dataset is not publicly open because it is a part of a project of collaboration. Some examples are shown in Figure 3.2. As shown in Figure 3.3, the microscopic images display the texture details inside the microorganism cells, so it is not surprising that many microscopic image classification algorithms rely on textural features. However, our optofluidic images (Figure 3.2) do not contain textural details inside the cells but only scattering patterns outside the cells. These appearance differences require new algorithms to be developed for optofluidic images.

FIGURE 3.3: Examples of images in the microscopic image dataset EMDS [40]

Image source: [Paper "Environmental Microorganism Data Set" \[40\]](#)

The images in our dataset can be divided into several parts (Table 3.1). The cell in the centre of our images appears as a black (or close to black) filled circle. Then, the scattering patterns outside the cell can be regarded as 2 parts: an inner brighter (close to white) part which looks like a circle or an ellipse, and an outer dimmer (grey) part that looks like rings or has some irregular patterns. Finally, the outside most part of the image is a black (or close to black) background. From an intuitive observation, the overall shape of class C images' scattering patterns looks closer to a circle than an ellipse for the other two classes, indicating that morphological (shape-related) descriptors may be useful for classification. The outer scattering patterns of different classes seem different, indicating that a feature representing the local appearances (greyscale intensity values) may also be useful. The observation on these images implies what kind of features to be considered for classification.

TABLE 3.1: Observation on image appearances of our dataset

Part	Appearance	Possible features for classification
Cell	A black (or close to black) filled circle in the centre	-
Scattering pattern, inner brighter part	Close to white, looks like a circle or an ellipse	Different overall shapes: Morphological (shape-related) descriptors
Scattering pattern, outer dimmer part	Grey, looks like rings or has some irregular patterns	Different overall shapes: Morphological (shape-related) descriptors Different intensity details: Features describing local appearances
Background	Black (or close to black)	-

3.2 Transmission Electron Microscope (TEM) Images of Virions

There are many image datasets of daily scenes and objects that are available for free, such as ImageNet [82] and PASCAL VOC [83]. Unlike these real-world images, the field of microorganism classification does not have many free, open-sourced datasets because capturing images for microorganisms need professional laboratory instruments and trained experts, and different research objectives require different types of images (e.g., colour/greyscale, single-cell/multiple-cell/colony, microscopic/optofluidic images of different microorganism species). Many images used for microorganism classification were collected by the research group itself and not open to the public [31],[41],[43]–[45],[47],[84]–[89]. Among these papers, some used the images of Petri dishes with bacterial colonies/virus plaque [84],[86],[88],[89] or spectroscopic images [85] (not direct imaging of the microorganisms), others all used microscopic images.

As summarized in Table 3.2, among the limited open-sourced microorganism datasets available online, the dataset BCO-DMO 670706 [23] was selected to verify the proposed algorithms for it contains single-cell greyscale images of 2 classes of microorganisms.

TABLE 3.2: Open-sourced microorganism datasets

Dataset	Description
SPL [90]	High-resolution electron microscopic images of viruses, bacteria, protozoa, and medical uses Not free of charge
Dataset in Dhindsa's paper [50]	Only the feature values and targets open to the public Extracted from microscopic images of 10 classes of microorganisms
EMDS [40]	Microscopic images of environmental microorganism (EM) 21 classes, 20 images per class Used in these microorganism classification studies [29],[30],[40],[46],[77] Colour images, some of them are single-cell
BCO-DMO 670706 [23]	Microscopic images taken by transmission electron microscope (TEM), 2 types/classes of coccolithophorid algae (<i>Pleurochrysis carterae</i>) virions Most are single-cell images

The images in the open-sourced dataset BCO-DMO 670706 [23] are microscopic greyscale images. Pre-processing was done to use the single-cell images with the same magnification. The details of pre-processing will be explained in Section 7.1. The image size after pre-processing is 1696*1696 pixels. The numbers of type 1 and type 2 images are 14 and 6 respectively. Some examples are shown in Figure 3.4. Unlike our optofluidic images, these images have a darker object (the microorganism in the centre) than the background. The background is grey and noisy, while the background in our images is black or nearly black and seems clearer.

FIGURE 3.4: Examples of images in the open-sourced dataset BCO-DMO 670706 [23] after pre-processing

Numbers of used image samples of Type 1, 2 are 14, 6 respectively.

Chapter 4

Entropy of Histogram of HOG for Waterborne Pathogen Classification

4.1 Introduction

With the aim, developing image classification algorithms for potential portable, real-time water monitoring applications, different features and classifiers were investigated on our dataset of waterborne pathogen images in this chapter. Some existing features were decided to be included in our classification algorithm because they were applicable for our images, simple to extract, and able to differentiate the classes. Besides, a new feature, the entropy of histogram of HOG (EHHOG) calculated based on the local edge descriptor HOG (histogram of oriented gradients), was proposed for classification. As in most classification algorithms, the testing accuracy was used as the performance evaluation metric [19],[38]. The proposed algorithm had a good classification accuracy while the computational cost was lower than the existing microorganism classification algorithms, showing the potential for portable, real-time applications.

4.2 Investigation on Various Features

Our dataset contains single-cell images of waterborne pathogens captured by an optofluidic system, which is cheaper, faster and smaller than conventional microscopes, thus more suitable for portable, real-time applications. Most of the existing microorganism classification algorithms were designed for microscopic images (as listed in Section 3.2). Thus, the existing features were investigated on whether they were applicable for our optofluidic images and

whether they could differentiate the classes of our dataset. The boxplots were used to evaluate the classification ability of the features.

4.2.1 Feature Category

As discussed in Chapter 2, there are various features used in existing methods for microorganism classification. Among these papers [42]–[44],[47],[49], the features can be categorized as in Table 4.1 with their extraction methods and characteristics. The categories are morphological/ geometric features, optical features, textual features, and others. These existing features used for microscopic images would be analyzed on whether suitable to classify our optofluidic image in Sections 4.2.2, 4.2.3.

Morphological features, or geometric features, are the features describing the shape of an object, usually extracted from the binarized images. They were commonly used for microorganism classification for many microorganism images have different object shapes from an observation from human eyes.

Optical features are derived from the histogram of an image. They were used in image classifications for statistical representations of the intensities.

Textual features are features describing the texture of an image, for example, features derived from the co-occurrence matrix of an image [91],[92]. Optical features and textural features are statistically calculated, so sometimes both referred to as statistical features.

Other features include features extracted with special pre-processing, features calculated by dimensionality reduction methods, and derived from transformed images. They are usually designed for specific images and not widely used.

TABLE 4.1: Features used in existing microorganism image classification algorithms

Category	Description	Examples of features in this category	Extraction method
Morphological/	Features to represent/	Area	Area of an object

geometric	describe the object shape, usually calculated from the segmented/ binarized images.	Aspect ratio	Ratio of the lengths of the major axis and minor axis of an ellipse (the ellipse that has the same second-moments as the object)
		Compactness	Ratio of an object's area and the area of a circle with the same perimeter of the object; It is referred to with different names, e.g., shape factor = roundness = $1/\text{compactness} = 1/\text{circularity}$
		Eccentricity	Ratio of the distance between the foci of the ellipse and its major axis length (the ellipse that has the same second-moments as the object)
		Fullness	Ratio of an object's area and the area of the rectangle bounding box of the object
		Perimeter	Perimeter of an object
		Solidity	Ratio of an object's area and the area of its convex hull
Optical	Features derived from the histogram of the image pixel values	Entropy	Entropy of the histogram
		Maximum	Maximum value of the histogram

	(intensities for greyscale images, RGB values for colour images). Sometimes called statistical features.	Maximum location	Maximum value location of the histogram (the intensity that occurs most frequently)
		Mean	Mean value of all intensities
		Minimum	Minimum value of the histogram
		Minimum location	Minimum value location of the histogram (the intensity that occurs least frequently)
		Percentiles	Percentile values of the histogram, e.g., the 10 percentile, 25 percentile (quartile), 50 percentile (median value), etc.
		Range	Maximum - minimum
		Variance	Variance of the histogram
Textual	Features to represent/ describe the texture of the images, e.g., features derived from the co-occurrence matrix. Sometimes called statistical features.	Contrast	Contrast of the co-occurrence matrix
		Correlation	Correlation of the co-occurrence matrix
		Entropy	Entropy of the co-occurrence matrix
		Homogeneity	Homogeneity of the co-occurrence matrix
		Maximum probability	Maximum value of the co-occurrence matrix
		Sum entropy	Sum entropy of the co-occurrence matrix

Others	Other features that cannot be categorized in above types, ususally designed for specific images and not widely used, like colour information of an RGB image, principal components (PC), and features derived from a transformed image, e.g., the local binary pattern (LBP) image, Log-Gabor filtered image, etc. .
--------	--

4.2.2 Boxplot Check

To preliminarily assess whether a feature has the potential to distinguish the 3 classes of our dataset, the boxplot was used to check whether the range of the feature value varies for different classes. All the features were normalized. If at least the boxes of 2 classes separate, i.e., the 25 percentile of one class's feature value is higher than the 75 percentile of another class's, this feature is regarded as capable to classify our images, which is called "passing the boxplot check". Here, the boxplots of the existing features that were considered in our algorithms were displayed in Figures 4.1, 4.2, and 4.3. The details of their calculation procedures are presented in Appendix F. The analysis of all the existing features summarized in Table 4.1 will be discussed in Section 4.2.3. The complete results of the boxplot check are shown in Appendices A, B.

In Figures 4.1, 4.2, the boxes of class G images and class C images are separate, meaning that the feature aspect ratio (AR) and solidity (SO) from Otsu binarized [24] images can pass the boxplot check.

FIGURE 4.1: Illustration of boxplot of the binarized-image-based feature AR, using the images in our dataset (training images of 1 fold)

AR: aspect ratio. AR_G, AR_C, AR_O refers to AR of class G, C, O images respectively.

FIGURE 4.2: Illustration of boxplot of the binarized-image-based feature SO, using the images in our dataset (training images of 1 fold)

SO: solidity. SO_G, SO_C, SO_O refers to SO of class G, C, O images respectively.

In Figure 4.3, all the 3 boxes overlap in terms of the feature value range, meaning that the compactness (CO) from Otsu binarized images cannot pass the boxplot check. It should be noted that CO from the binarized images using the proposed Multi-Region binarization method can pass the boxplot check in some cases, so the feature CO is also considered in the later algorithm (to be elaborated in Chapter 5).

FIGURE 4.3: Illustration of boxplot of the binarized-image-based feature CO, using the images in our dataset (training images of 1 fold)

CO: compactness. CO_G, CO_C, CO_O refers to CO of class G, C, O images respectively.

The features AR, SO, CO was extracted from the Otsu binarized images where only the largest object was kept. In the classification experiments, the training and testing samples were partitioned using 10-fold cross-validation (CV). Only the results of the training samples were used for decision according to the machine learning concept.

4.2.3 Analysis on Features of Different Categories

The features were analyzed based on the aim of developing microorganism image classification algorithms that have the potential for portable, real-time applications, so the features to be used should meet the following requirements.

- Simple to extract

- Capable to classify the 3 classes of our optofluidic image dataset, i.e., can pass the boxplot check, preferably with different numbers of samples per class
- Small total number of features
- Better to have comprehensible physical meanings so that easy to decide whether to use/adapt this feature when extending to other applications

Morphological Features

Morphological features were usually extracted from the binarized images. Binarization converts an image into black and white, corresponding to the background and the object (foreground). Otsu's method [24] is conventional, fast, and commonly used for binarization. Otsu binarization was used in the experiments if not specified otherwise. More binarization methods will be discussed in Chapter 5. The morphological features have clear physical meaning, i.e., describing the shape of an object. For example, in our dataset, the object in class G images looks like an ellipse while the object in class C images looks like a circle, so the box of the aspect ratio (the ratio between the major and minor axis lengths) of class G images is higher than that of class C images (Figure 4.1). Another advantage of the morphological features is that they are often simple to extract. Thus, morphological features are very common in microorganism image classification, the existing methods [29],[30],[41]–[47],[49],[50] all included morphological features.

Combining the discussion in Section 4.2.2 and Table 4.2, the aspect ratio (AR) and solidity (SO) were 2 morphological features that preliminarily chosen for our classification algorithm. Their ability to distinguish the 3 classes of our dataset was also proved in the classification experiments.

TABLE 4.2: Boxplot check results of some morphological features on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Feature name	Boxplot check result	Note
Area	1	Passed for the original dataset, but did not pass the boxplot check for the datasets with 30, 70, 100, 300, 700, 1000 images per class
Aspect ratio	1	Always passed the boxplot check
Eccentricity	0	
Fullness	0	
Minor axis length	0	
Perimeter	0	
Solidity	1	Often passed the boxplot check

Optical Features

Optical features are derived from the histogram of an image. They have the physical meaning representing the image intensity (or RGB values for colour images) distributions, but the classification ability usually varies for different datasets. For example, the optical features worked well for classifying images in [43] could not pass the boxplot check for our images (Table 4.3).

Besides, many optical features can be extracted from the histogram, but it is hard to tell which or which combination of them can perform well for the image classification. For example, the 25 percentile, 50 percentile of the image histogram of our dataset can pass the boxplot check, but the 10 percentile, 75 percentile, and 99 percentile cannot (Table 4.3), indicating that among 1 percentile, 2 percentile, ..., 99 percentile, there is no rule to select the feature(s) for classification, features worked for other people's dataset may not work for our dataset.

TABLE 4.3: Boxplot check results of some optical features on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Feature name	Boxplot check result	Note
Mean	0	Mean, 10 percentile, and 99 percentile are features used in [43] Not able to predict which percentile will be useful in classification
10 percentile	0	
99 percentile	0	
25 percentile	1	
50 percentile	1	
75 percentile	0	
Entropy	0	

Textual Features

Some microscopic images show the texture inside the microorganism cell, so the textual features can be useful for their classification [49]. But for our optofluidic images, the structure inside the cell is not shown. And similar to optical features, it is hard to select the feature(s) to be used among a large number of features. For example, in Bueno's paper [49], 19 features were calculated from the co-occurrence matrix, and the co-occurrence matrix was calculated with 3 different distances (1, 3, 5 pixels) and 4 different directions (0°, 45°, 90°, 135°), so there are $19 \times 3 \times 4 = 228$ features related to the co-occurrence matrix. Most of them cannot pass the boxplot check for our images with different numbers of samples (see the last column of the 21st feature to the 248th feature in Table B.1).

Other Features

Other features that have been investigated are also not very useful for our classification objective. Some cannot pass the boxplot check (Table 4.4); some require complicated calculation and thus not very suitable for the aim of future portable, real-time applications (e.g., image transformation); some cannot be extracted from our greyscale images (e.g., colour information); some need to be extracted from a batch of samples (e.g., principal component analysis

(PCA) [93],[94], non-negative matrix factorization (NMF) [95],[96]), thus slow, not very reliable with a small number of images, and cannot test a single image. Also, for the dimensionality reduction techniques like PCA and NMF often applied to reduce the dimensions of the features, there is no way to predict how many features transformed and reduced from the original features can achieve the best classification performance, so a large number of trials of training and testing may be needed.

TABLE 4.4: Boxplot check results of some other features on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Feature name	Boxplot check result	Note
Hu moment 1	0	Features used in [42],[49] Not able to predict which moment will be useful in classification Complicated to extract
Hu moment 2	1	
Hu moment 3	0	
Hu moment 4	0	
Hu moment 5	0	
Hu moment 6	0	
Hu moment 7	0	

4.2.4 Feature Investigation Summary

The complete boxplot check results of the features investigated are displayed in Appendices A, B. Since the features used in the papers [42],[44],[47] were covered by the features in Kumar's method [43] and Bueno's method [49], Appendices A, B only include the features in Kumar's method and Bueno's method.

Based on previous discussions, there were many existing features used for microorganism image classification, but the existing algorithms were developed for microscopic images so the existing features may not be suitable to classify our optofluidic images. Many of them did not pass the boxplot check (Tables 4.2, 4.3, 4.4, B.1), meaning that these features did not have different value ranges for different classes. Among the morphological features, only the aspect ratio (AR) and solidity (SO) passed the boxplot check under different cases of dataset sizes. The optical features and textural features were not chosen for most of them did not pass the boxplot check and there was no way to predict which are the several ones that may be

useful. Besides, the microscopic images show the texture inside the microorganism cells, so textural features were often used. But for our optofluidic images, there is no texture inside the cell, so textural features were not very useful for classification according to the boxplot check results. The other features were not considered for reasons like not passing the boxplot check, being too complicated to calculate, or not reliable on small datasets. Finally, AR and SO from Otsu binarized images were preliminarily chosen for the proposed classification algorithm.

4.3 Proposed Feature: Entropy of Histogram of Histogram of Oriented Gradients (EHHOG)

4.3.1 Histogram of Oriented Gradients (HOG)

The histogram of oriented gradients (HOG) [97] of an image has the horizontal axis showing the orientation bins of image gradients, and the vertical axis showing the weighted voting values of corresponding orientation. With many image processing applications, like W.T. Freeman and M. Roth's hand gesture recognition algorithm in 1995 [98], and N. Dalal and B. Triggs's human detection algorithm in 2005 [99], HOG has become a frequently used feature to describe local edge appearances. In the field of microorganism image processing, there are also applications of HOG, for example, it can be used for automatic segmentation of virus plaques [88]. The calculation procedures of HOG are as follows.

Partial Derivatives

A greyscale image can be represented by a function $f(i, j)$ with the pixel value (i.e., intensity/ grey level) f at the pixel location (i, j) along x (horizontal), y (vertical) direction. The partial derivatives along the x, y directions are defined by Equations (4.1), (4.2) respectively.

$$g_x = \frac{\partial f}{\partial x} \quad (4.1)$$

$$g_y = \frac{\partial f}{\partial y} \quad (4.2)$$

They can be approximated using 1-D filters with convolutions. (Equations (4.3), (4.4))

$$g_x = [-1 \quad 0 \quad +1] \circledast f \quad (4.3)$$

$$g_y = \begin{bmatrix} -1 \\ 0 \\ +1 \end{bmatrix} \circledast f \quad (4.4)$$

Here, $\circledast$ denotes the convolution operation.

With the definition of convolution, the partial derivatives are:

$$\begin{aligned} g_x(i, j) &= 1 \times f(i-1, j) + 0 \times f(i, j) + (-1) \times f(i+1, j) \\ &= f(i-1, j) - f(i+1, j) \end{aligned} \quad (4.5)$$

where $i = 1, 2, \dots, m; j = 1, 2, \dots, n$

$$\begin{aligned} g_y(i, j) &= 1 \times f(i, j-1) + 0 \times f(i, j) + (-1) \times f(i, j+1) \\ &= f(i, j-1) - f(i, j+1) \end{aligned} \quad (4.6)$$

where $i = 1, 2, \dots, m; j = 1, 2, \dots, n$

Here, m denotes the image width, n denotes the image height (in pixels). For pixel values outside the image, 0 is usually used for padding.

Image Gradient

Mathematically, the image gradient is defined by Equation (4.7).

$$\nabla f(i, j) = \begin{bmatrix} g_x(i, j) \\ g_y(i, j) \end{bmatrix} \quad (4.7)$$

where $i = 1, 2, \dots, m; j = 1, 2, \dots, n$

Thus, the magnitude of the image gradient is given by Equation (4.8).

$$M(i, j) = \sqrt{[g_x(i, j)]^2 + [g_y(i, j)]^2} \quad (4.8)$$

where $i = 1, 2, \dots, m; j = 1, 2, \dots, n$

The orientation (i.e., direction) of the image gradient is given by Equation (4.9).

$$\theta(i, j) = \tan^{-1} \left[\frac{g_y(i, j)}{g_x(i, j)} \right] \quad (4.9)$$

where $i = 1, 2, \dots, m; j = 1, 2, \dots, n$

Here, m denotes the image width, n denotes the image height (in pixels). The orientation θ has the range of $[0^\circ, 180^\circ)$.

Orientation Discretization

After computing the image gradient, the orientations are discretized into 9 bins: $[0^\circ, 20^\circ)$, $[20^\circ, 40^\circ)$, $[40^\circ, 60^\circ)$, $[60^\circ, 80^\circ)$, $[80^\circ, 100^\circ)$, $[100^\circ, 120^\circ)$, $[120^\circ, 140^\circ)$, $[140^\circ, 160^\circ)$, $[160^\circ, 180^\circ)$.

The orientation bin of each pixel can be represented by Equation (4.10).

$$\delta(t, (i, j)) = \begin{cases} 1 & \text{if } t = \left\lfloor \frac{\theta(i, j)}{20^\circ} + 1 \right\rfloor \\ 0 & \text{if } t \neq \left\lfloor \frac{\theta(i, j)}{20^\circ} + 1 \right\rfloor \end{cases} \quad (4.10)$$

where $t = 1, 2, \dots, 9$

Here, t denotes the 9 orientation bins.

Weighted Voting Values

Pixels are grouped into cells (e.g., 1 cell = 8×8 pixels), and each pixel contributes to the voting values of its orientation bin with its magnitude (i.e., weighted voting values) on the cell histogram. The cell histogram can be represented by Equation (4.11):

$$h_{\text{Cell}}(t) = \frac{1}{\sum_{\text{all } (i, j) \text{ in the Cell}} [M(i, j)]} \sum_{\text{all } (i, j) \text{ in the Cell}} [\delta(t, (i, j)) \cdot M(i, j)] \quad (4.11)$$

where $t = 1, 2, \dots, 9$

Here, t denotes the 9 orientation bins.

Cells are grouped into blocks (e.g., 1 block = 2×2 cells, with 50% overlapping between 2 blocks), then combines to be the HOG (histogram of oriented gradients) of the whole image. The details of HOG formation (how pixels grouped into cells and blocks to form the HOG of the whole image) are explained in Appendix C.

4.3.2 Entropy of Histogram of Histogram of Oriented Gradients (EHHOG)

The 2 existing features preliminarily chosen for classification (aspect ratio (AR) and solidity (SO)) are morphological features extracted from binarized images, and the existing features extracted from greyscale images were not suitable so not considered (Section 4.2). Through an intuitive observation of our images, different classes have different scattering patterns. During the binarization process, these details of greyscale intensity information may be lost. Thus, we proposed a feature, the entropy of histogram of HOG (EHHOG), extracted from greyscale images to describe the local appearances of the image.

Calculation Procedures

EHHOG is derived by discretizing HOG (histogram of oriented gradients) and computing its entropy as shown in Figure 4.4 and Equation (4.12). For HOG, the horizontal axis is the orientation bins, and the vertical axis is the weighted voting values, i.e., HOG values (denoted as h). For the histogram of HOG, the horizontal axis is the HOG values (denoted as h), and the vertical axis is the probability of the value existing in the HOG (denoted as p). The details of calculating EHHOG are explained in Appendix C.

$$EHHOG = - \sum_{all\ p} p(\log p) \quad (4.12)$$

FIGURE 4.4: Flow chart of the procedures to extract EHHOG

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients.

Boxplot Check

From the boxplot of EHHOG (e.g., Figure 4.5) on the greyscale images in our dataset, it can be concluded that the EHHOG has the ability to classify our images. It can pass the boxplot check for distinguishing class G and the other two classes in terms of the feature value. Especially, the boxes of class G and class O have quite different ranges, and in the boxplots of the features AR (aspect ratio), SO (solidity) the boxes of class G and class O always overlap a lot, indicating that adding EHHOG may be very useful for classification.

FIGURE 4.5: Illustration of boxplot of the proposed feature, EHHOG, using our dataset's images (original greyscale intensity) (training images of 1 fold)

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. EHHOG_G, EHHOG_C, EHHOG_O refers to EHHOG of class G, C, O images respectively.

The EHHOG could also be extracted from the binarized image, but it did not pass the boxplot check (Figure 4.6). This is consistent with the concept that HOG is to represent the local appearances, and binarized images have lost much information of the local details.

FIGURE 4.6: Illustration of boxplot of the proposed feature, EHHOG, using our dataset's images (Otsu binarized) (training images of 1 fold)

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. EHHOG_G, EHHOG_C, EHHOG_O refers to EHHOG of class G, C, O images respectively.

Visualization of HOG

For a better understanding of the HOG (histogram of oriented gradients) and the proposed feature EHHOG (entropy of histogram of HOG), an example of HOG visualization is shown in Figures 4.7, 4.8. As explained in Section 4.3.1, every 8*8 pixel patch forms a cell on the image, and each star-like green mark in Figures 4.7, 4.8 corresponds to a cell histogram. The cell histogram has the horizontal axis of orientation bins, represented by the bar directions of each green "star"; the vertical axis of weighted voting values (denoted as h), represented by the bar lengths of each green "star" on the figures.

FIGURE 4.7: An example of visualization of HOG

HOG: histogram of oriented gradients.

FIGURE 4.8: An example of visualization of HOG, zoomed in

HOG: histogram of oriented gradients.

Visualization of Histogram of HOG

The cell histograms are combined to be the HOG (histogram of oriented gradients) of the whole image. Then the histogram of HOG will have the horizontal axis of HOG values (denoted as h), and the vertical axis of the probability of the value existing in the HOG (denoted as p) as shown in Figures 4.9, 4.10, and 4.11. The entropy calculation on the histogram of HOG gives EHHOG (entropy of histogram of HOG).

FIGURE 4.9: An example of HOG, histogram of HOG of a class G image

HOG: histogram of oriented gradients.

FIGURE 4.10: An example of HOG, histogram of HOG of a class C image

HOG: histogram of oriented gradients.

FIGURE 4.11: An example of HOG, histogram of HOG of a class O image

HOG: histogram of oriented gradients.

Figures 4.12, 4.13, and 4.14 show the distributions of the histogram of HOG (histogram of oriented gradients) for class G, C, O respectively, and Figure 4.15 shows the mean values of the 3 classes. In Figure 4.15, it can be observed that the curve of class G images' mean histogram of HOG is distinguishable. This is consistent with Figure 4.5 that the feature EHHOG (entropy of histogram of HOG) will be very useful in classification for its ability to differentiate class G and class O, which is not applicable for other features.

FIGURE 4.12: Distribution of the histogram of HOG, class G

HOG: histogram of oriented gradients.

FIGURE 4.13: Distribution of the histogram of HOG, class C

HOG: histogram of oriented gradients.

FIGURE 4.14: Distribution of the histogram of HOG, class O

HOG: histogram of oriented gradients.

FIGURE 4.15: Distribution of the histogram of HOG, mean values of 3 classes

HOG: histogram of oriented gradients.

In information theory, a larger entropy corresponds to a more uniform histogram distribution. For a single image, a larger EHHOG (entropy of HOG) means more uniform distribution of p (p is the probability of HOG (histogram of oriented gradients) value h), i.e., the histogram of HOG of this image is a flatter curve.

If we observe the green “stars” in Figure 4.8, we can see that if all the bar lengths in a cell have the same length (uniform distribution of h), there is no edge; if 2 or 3 bars are very long and other bars are very short (a nonuniform distribution of h), there is a clear edge along the direction of the long bars. A uniform distribution of h of the cell histogram contributes to nonuniformly distributed p , and a non-uniform distribution of h of the cell histogram contributes to less nonuniformly distributed p (not necessarily uniform) of the histogram of HOG of the whole image. Thus, EHHOG represents whether an image has more “non-edge” cells or more “edge” cells. It summarizes the local edge appearances of a whole image. This physical meaning cannot be directly observed on the images by human eyes but is comprehensible with the mathematical derivation. Similar to Section 4.2, considering the boxplot check result, the simple calculation steps and the comprehensible physical meaning of EHHOG, this feature would be included in the classification algorithm for our waterborne pathogen images.

4.4 Classification Results and Discussions

In this thesis, all the classification methods were evaluated with 10-fold cross-validation (CV). The experiments of handcrafted-feature-based methods were performed on a workstation with an Intel Core i7-4770 CPU with MATLAB. The experiments of deep-learning-based methods were performed on an Intel Xeon CPU E5-2630 workstation with NVIDIA TITAN Xp GPU with Python. The settings of the models not specified were keeping the default values in MATLAB/Python.

4.4.1 Using Different Classifiers

The support vector machines (SVM), decision trees (DT)/ bagging decision trees (BDT), and artificial neural networks (ANN) are commonly used classifiers in the field of image classification [19],[37],[39],[40]. Especially, the SVM was identified as the most popular classifier in the literature review in the papers by S. Kosov, K. Shiraham, et al. [40] and C. Li, K. Wang, et al. [19]. Based on previous investigations and discussions in Sections 4.2, 4.3, the features from greyscale images and binarized images were extracted for our dataset (Table 4.5).

TABLE 4.5: Feature extracted from greyscale images and binarized images

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity.

Feature number	Feature name	Extracted from
1	EHHOG	Greyscale images
2	AR	Binarized images (Otsu's method)
3	SO	

Support Vector Machine and Bagging Decision Trees

The classification results using the support vector machine (SVM) or bagging decision trees (BDT) as the classifier are summarized below.

TABLE 4.6: Classification results of using features in Table 4.5 with SVM and BDT on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. SVM: support vector machine. Linear SVM: SVM with linear kernel. RBF SVM: SVM with RBF (radial basis function) kernel. BDT: bagging decision trees. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Feature set	Testing accuracy		
	Linear SVM	RBF SVM	BDT
1	61.9%	74.9%	63.2%
2	60.9%	62.4%	53.7%
3	45.6%	54.4%	41.7%
1, 2	73.0%	79.8%	73.9%
1, 3	77.0%	87.1%	82.9%
2, 3	66.5%	77.8%	71.5%
1, 2, 3	80.6%	91.7%	92.2%

The best result using linear support vector machine (SVM) was 80.6% with all the 3 features (i.e., feature set "1, 2, 3"), much lower than using radial basis function (RBF) SVM or bagging decision trees (BDT), so linear SVM is not a good choice here. In many cases, the RBF SVM gave higher accuracies than BDT, but the best accuracy of 92.2% came from the feature set "1, 2, 3" with BDT. The accuracy of the feature set "1, 2, 3" with RBF SVM was 91.7%, slightly lower than 92.2%, so both RBF SVM and BDT are tested in later experiments.

From Table 4.6, it can be observed that the accuracies using the 2-D feature sets (i.e., feature sets "1, 2", "1, 3", and "2, 3") were higher than those using a single feature (i.e., feature sets "1", "2", and "3"), and the accuracy using all 3 features was the best among all feature sets. This supported our idea of combining the features extracted from binarized images and greyscale images for classification. And the results after the combination of features may not have the same trend as using a single feature, for example, the accuracy of using the feature set "1, 3" was higher than using "1, 2" but the accuracy using feature "3" was much lower than using "2". As discussed in Sections 4.2, 4.3, AR (aspect ratio) and SO (solidity) describe the object shape of a binarized image, and EHHOG (entropy of histogram of histogram of oriented gradients) describes the local appearances of a greyscale image. The boxplots (Figures 4.1, 4.2, 4.5) also show that the abilities to distinguish the 3 classes were different for features from binarized images and from greyscale images. For the features extracted from binarized images, class C was differentiated from the other two classes, while for the features extracted from greyscale images, class G was differentiated from the other two classes on the boxplot. In this way, they extract different types of information from the image, so features from binarized images and features from greyscale images complement each other in classification. Thus, the proposed algorithm (called EHHOG method or EHHOG algorithm later) uses AR, SO, and EHHOG together for waterborne pathogen image classification.

Artificial Neural Networks with Different Configurations

The artificial neural network (ANN) can have infinite configurations in theory with different numbers of hidden layers and different numbers of neurons in each hidden layer. Considering the limitation of time and computational power, the following configurations were examined.

- 1) One hidden layer
 - a) 1 neuron per hidden layer
 - b) 2 neurons per hidden layer
 - c) ...
 - d) 300 neurons per hidden layer
- 2) Two hidden layers
 - a) 1 neuron per hidden layer

- b) 2 neurons per hidden layer
 - c) ...
 - d) 120 neurons per hidden layer
- 3) Three hidden layers
- a) 1 neuron per hidden layer
 - b) 2 neurons per hidden layer
 - c) ...
 - d) 70 neurons per hidden layer
- 4) Four hidden layers
- a) 1 neuron per hidden layer
 - b) 2 neurons per hidden layer
 - c) ...
 - d) 50 neurons per hidden layer
- 5) Five hidden layers
- a) 1 neuron per hidden layer
 - b) 2 neurons per hidden layer
 - c) ...
 - d) 35 neurons per hidden layer

The results are shown in the figures and the table below.

FIGURE 4.16: Classification results of using features EHHOG, AR, SO with 1-hidden-layer ANN on our dataset

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. ANN: artificial neural network. The marker shows the index and value of the highest accuracy.

FIGURE 4.17: Classification results of using features EHHOG, AR, SO with 2-hidden-layer ANN on our dataset

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. ANN: artificial neural network. The marker shows the index and value of the highest accuracy.

FIGURE 4.18: Classification results of using features EHHOG, AR, SO with 3-hidden-layer ANN on our dataset

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. ANN: artificial neural network. The marker shows the index and value of the highest accuracy.

FIGURE 4.19: Classification results of using features EHHOG, AR, SO with 4-hidden-layer ANN on our dataset

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. ANN: artificial neural network. The marker shows the index and value of the highest accuracy.

FIGURE 4.20: Classification results of using features EHHOG, AR, SO with 5-hidden-layer ANN on our dataset

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. ANN: artificial neural network. The marker shows the index and value of the highest accuracy.

TABLE 4.7: Classification results of using features EHHOG, AR, and SO with ANN on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. ANN: artificial neural network. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Number of hidden layers	Highest testing accuracy	
	Accuracy	Number of neurons per hidden layer
1	91.7%	40
2	91.5%	32
3	88.3%	17
4	85.9%	12
5	88.8%	11

Table 4.7 summarizes the highest accuracy and the corresponding number of hidden neurons of the artificial neural network (ANN) with certain numbers of hidden layers. The 5 rows of the table are related to Figures 4.16, 4.17, 4.18, 4.19, and 4.20 respectively. It can be

observed that although the best accuracy with ANN (91.7%, 1 hidden layer, 40 neurons per hidden layer) was close to the accuracy with bagging decision trees (BDT) as the classifier (92.2%), the accuracies with different ANN configurations were very different and there was no rule to predict where the best accuracy would exist. That means that if ANN was used, hundreds or even thousands of times of classifier training and testing were needed to pick the best configuration, while if BDT or support vector machine (SVM) was used, there was only one time of training and testing. ANN had much higher requirements on time and computational power in training and testing but did not improve the accuracy than BDT or SVM. Thus, ANN was not used in later experiments.

4.4.2 Using Existing Microorganism Classification Methods on Our Dataset

The proposed EHHOG method using the proposed feature EHHOG (EHHOG: entropy of histogram of HOG) (HOG: histogram of oriented gradients) and 2 morphological features AR (AR: aspect ratio), SO (SO: solidity) was compared with some existing microorganism classification methods. The papers by S. Kumar and G.S. Mittal [43], G. Bueno, et al. [49], and A. Pedraza, et al. [73] were selected for the following reasons:

- Their microorganism classification algorithms were used for comparison/ reference purposes in many similar studies.
- Their images were also single-cell greyscale images of microorganisms.

The main difference between our images and the images used in these papers is that our images were captured by an optofluidic system and their images were captured by conventional microscopes.

The paper by S. Kumar, and G.S. Mittal [43] proposed 9 features (details in Appendix A) and an artificial neural network (ANN) with one hidden layer to classify 5 types of bacterial images. The number of neurons in the hidden layer is given by Equation (4.13). Using these 9 features to classify our images with this ANN, the testing accuracy was only 82.7%. If radial basis function (RBF) support vector machine (SVM) or bagging decision trees (BDT) were used as the classifier, the accuracy could be higher, so we used the results with SVM and BDT for later comparison.

$$N_h = \frac{1}{2} (N_f + N_c) + \sqrt{N_t} \quad (4.13)$$

Here, N_f denotes the number of features, N_c denotes the number of classes, and N_t denotes the number of training samples.

The paper by G. Bueno, et al. [49] developed an algorithm that extracted 1460 handcrafted features, selected features (about 100 or 200 features selected by the criterion described in [100]), and classified their 80 types of diatom images using bagging decision trees (BDT) with 200 learners (trees).

The paper by A. Pedraza, et al. [73] was also from the VISILAB Research Group in the University of Castilla La Mancha, but classifying these diatom images using a pre-trained deep learning model (pre-trained AlexNet [72]).

Deep learning is a kind of machine learning approach that combines the feature extractor and the classifier in one single model and does not require handcrafted features and a separate classifier. It does not require manual investigation to decide the features like the handcrafted-feature-based methods but requires much higher computational power for training and testing. For example, Pedraza's method required more than 15 times of computational complexity of the proposed method in terms of the number of floating-point operations (FLOPs) for testing (Table 4.8). Besides, the computational cost of training for a deep learning model is much heavier than training a classifier with some handcrafted features. Training a deep learning model requires a GPU (graphics processing unit) or even multiple GPUs. Training the handcrafted-feature-based models used in our experiments only requires a workstation with a CPU (central processing unit). These handcrafted-feature-based models are even possible to be trained on a laptop, which is cheap and light. Thus, deep learning is a popular method for image classification but not very suitable for the aim of future portable, real-time water monitoring applications, especially in rural or undeveloped areas. The settings for applying the deep learning model to our dataset are presented in Appendix E. The detailed explanation and calculation method of the number of FLOPs are presented in Appendix G.

TABLE 4.8: Classification results of comparing the proposed method and existing methods on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. FLOP: floating-point operation, multiply-accumulate operation considered, M: $\times 10^6$. Computational complexity for a forward pass (i.e., testing one image) in terms of number of FLOPs. SVM: support vector machine. Linear SVM: SVM with linear kernel. RBF SVM: SVM with RBF (radial basis function) kernel. BDT: bagging decision trees. * denotes the proposed method. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Method	Number of features	Computational complexity	Testing accuracy with different classifier		
			Linear SVM	RBF SVM	BDT
EHHOG method *	3	41 MFLOPs	80.6%	91.7%	92.2%
Kumar's method [43]	9	98 MFLOPs	75.0%	89.5%	86.1%
Bueno's method [49]	255	61 MFLOPs	93.0%	41.3%	85.4%
Pedraza's method [73]	4096	716 MFLOPs	93.9%		

The accuracy of using Kumar's method [43] on our dataset was lower than the proposed EHHOG method with either support vector machine (SVM) or bagging decision trees (BDT), and the computational complexity in terms of the number of floating-point operations (FLOPs) was even larger than EHHOG method. Thus, this existing method [43] were not used for later comparisons in Chapters 5, 6.

With BDT as the classifier, the proposed EHHOG method outperformed the existing handcrafted-feature-based methods [43],[49], but not as good as Bueno's method [49] with linear SVM and the deep-learning-based Pedraza's method [73]. Considering the small number of features and low computational complexity, the EHHOG method is still a practical solution to waterborne pathogen classification with the aim of possible portable, real-time applications in the future.

4.5 Summary of Proposed EHHOG Algorithm

With the aim of developing waterborne pathogen classification algorithms that have the potential for portable, real-time applications, the features used in existing researches were categorized and investigated. Considering the extraction procedures, the boxplot check results, and the physical meanings, two morphological features, aspect ratio (AR) and solidity (SO), were chosen for the proposed algorithm. They describe the shape of an object and are extracted from the binarized images. (Section 4.2)

To better utilize the detailed information in the greyscale intensities of the image, a new feature named entropy of histogram of HOG (EHHOG) was proposed. It is calculated based on a commonly used local edge descriptor, the histogram of oriented gradients (HOG). Through the calculation steps, boxplots, and visualizations of EHHOG, the physical meaning of the feature and its ability to distinguish different classes in our optofluidic images of waterborne pathogens was clear and comprehensible. (Section 4.3)

Comparing the classification results of using features extracted from binarized images (AR, SO) and from greyscale images (EHHOG) separately and together, the accuracy using AR, SO, and EHHOG together was the highest, verifying our idea of combining these features. The proposed EHHOG algorithm, i.e., using the 3 features EHHOG, AR, and SO with bagging decision trees (BDT) as the classifier, achieved an accuracy of 92.2%, higher than using the existing methods [43],[49] with BDT. Although the accuracy was lower than Bueno's method [49] with linear support vector machine (SVM) and deep-learning-based Pedraza's method [73], the proposed algorithm has a much smaller number of features and requires less computational power. The proposed EHHOG algorithm also performed well for the augmented datasets with different numbers of samples (to be discussed in Section 6.4.2). (Section 4.4)

Based on the proposed feature EHHOG and the idea of combining features from binarized images and from greyscale images, more algorithms are to be developed and presented in Chapters 5, 6 to improve the classification accuracy while keeping the computational cost low.

Chapter 5

Multi-Region Binarization to Improve Waterborne Pathogen Classification

5.1 Introduction

Chapter 4 presented the proposed EHHOG method combining features extracted from the binarized images (aspect ratio (AR) and solidity (SO)) and the proposed feature EHHOG (entropy of HOG) (HOG: histogram of oriented gradients) extracted from the greyscale images. The classification accuracy with bagging decision trees (BDT) achieved 92.2%, which is a good performance considering the algorithm's low computational cost although lower than the existing Bueno's method [49] with linear support vector machine (SVM) or Pedraza's deep-learning-based method [73]. In this chapter, different binarization methods were investigated and a new Multi-Region Binarization method was proposed to capture the scattering patterns missed by the existing binarization methods. Experiments showed that using the proposed Multi-Region Binarization and a new feature combination $G1B1B\alpha$, the classification accuracy could be improved from EHHOG method while keeping the computational cost low.

5.2 Proposed Multi-Region Binarization Method

In image signal processing, binarization is a pre-processing technique that can improve classification performance [30],[53]. In the EHHOG method, the images are binarized by Otsu's method [24]. It is a conventional, fast, and commonly used binarization method. Otsu binarization uses a global threshold, meaning that the threshold is the same for all pixels in an image. In this case, there could be information missed when using Otsu's binarization because our images have different brightness at different parts of the image.

5.2.1 Multi-Region Binarization

By observing the images in our dataset (e.g., Figures 3.2, 5.2), it can be seen that they were all brighter (closer to white, related to large intensity value) in the central part and gradually became dimmer (closer to black, related to small or 0 intensity value) in the outer part. These variations in different local areas indicate that the global thresholding binarization using a single Otsu threshold everywhere in the image may not be good enough. According to this characteristic, we proposed a local thresholding image binarization method, Multi-Region Binarization, to extract the information that is missed by Otsu's method.

The input image is partitioned into multiple regions (R denotes the number of regions) with different binarization thresholds. The height and the width of the image are equally divided into $2R$ parts, then the outermost band is Region 1, the second outermost band is Region 2, ..., and the innermost rectangle is Region R as shown in Figure 5.1. The threshold of each region is defined as Equation (5.1).

$$Th_i = \frac{i}{R^2} \times (\text{Otsu's threshold}) \quad (5.1)$$

where $i = 1, 2, \dots, R$

Here, Th_i denotes the threshold of region i , R denotes the total number of regions.

Figure 5.1 shows the thresholds of the proposed Multi-Region Binarization with 3 regions. The thresholds become smaller from the central part to the outer part, corresponding to brighter (closer to white, related to larger intensity value) central part and dimmer (closer to black, related to small or 0 intensity value) outer part of the image.

FIGURE 5.1: Illustration of Multi-Region Binarization (number of regions = 3)

5.2.2 Comparing Binarized Images with Otsu and Multi-Region Binarization Method

Below shows an example of an image in our dataset and its binarization results by Otsu's method [24] and the proposed Multi-Region Binarization method. The optofluidic image has no texture inside the microorganism cell, and the cell/ particle appears as a round-like black (or close to black) area at the innermost part of the image. The scattering patterns outside the cell can be regarded as 2 parts: an inner brighter (close to white) part that looks like a circle or an ellipse, and an outer dimmer (grey) part that looks like a ring or has some irregular patterns (Figure 5.2). Using Otsu's method, only the inner bright part was binarized out (Figure 5.3), while using Multi-Region Binarization method, the outer part was also kept (Figure 5.4). This confirms the presumption in Section 5.2.1 that the proposed Multi-Region Binarization method can extract the information missed by Otsu's method.

FIGURE 5.2: Image example: original image of class O, #42 image

FIGURE 5.3: Image example: Otsu binarized image of class O, #42 image

FIGURE 5.4: Image example, Multi-Region binarized image of class O, #42 image

For Multi-Region Binarization, 2 to 6 regions were used in the experiments so that the binarized images have a clear object showing the scattering patterns. (image examples shown in Section 6.3.1)

5.3 Feature Combination

In EHHOG method (Chapter 4), the features extracted from the binarized images, aspect ratio (AR) and solidity (SO), and the proposed feature EHHOG (entropy of HOG) (HOG: histogram of oriented gradients) extracted from the greyscale images were used for classification on our optofluidic waterborne pathogen images. After proposing the Multi-Region Binarization method, more binarized images can be used for feature extraction. Besides, more greyscale images can be generated by filtering the original greyscale images. Thus, more features were obtained, and how to combine these features for classification should be investigated.

5.3.1 Features from Binarized Images

As discussed in Section 4.2.2, the feature compactness (CO) extracted from Otsu binarized images did not pass the boxplot check, so not included in the previous EHHOG method. However, CO from Multi-Region binarized images could pass the boxplot check sometimes (to be discussed in Chapter 6), so the 3 morphological features representing object shapes, AR, SO, CO, were all extracted from the binarized images (only largest object kept) here. Then the features extracted from Otsu [24] binarized images, Multi-Region binarized images with 2, 3, 4, 5, 6 regions that passed the boxplot check were named as the feature sets $B1$,

$B2$, $B3$, $B4$, $B5$, $B6$ respectively as in Table 5.1. The boxplot check results are summarized in Table 5.2. For example, there were 2 features, AR and SO from Otsu binarized images, in the feature set $B1$ according to the boxplots in Figures 4.1, 4.2, and 4.3.

TABLE 5.1: Features extracted from binarized images

AR: aspect ratio. SO: solidity. CO: compactness.

Binarization method	AR	SO	CO	Features passed the boxplot check
	Extracted from			
1	Otsu binarized images			B1
2	Multi-Region binarized images, 2 regions			B2
3	Multi-Region binarized images, 3 regions			B3
4	Multi-Region binarized images, 4 regions			B4
5	Multi-Region binarized images, 5 regions			B5
6	Multi-Region binarized images, 6 regions			B6

TABLE 5.2: Boxplot check results of features extracted from binarized images on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. AR: aspect ratio. SO: solidity. CO: compactness. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result		
	AR	SO	CO
1	1	1	0
2	1	1	0
3	1	1	0
4	1	0	0
5	1	0	0
6	1	0	0

5.3.2 Features from Greyscale Images

From human eyes' observation, the background (the outmost dark part, outside the scattering patterns) of our images seems black for all classes. To investigate whether the whole image or

only some central intensities of the greyscale images were useful for classification, different types of greyscale intensities were used to extract the feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) as below. The intensity filtering was based on the observation discussed in Section 5.2.2 that the scattering pattern could be regarded as a brighter, round-like inner part and a dimmer, ring-like or irregular outer part.

- The intensity of the original greyscale images
- The intensity of the image filtered by the difference of the Multi-Region binarized image and Otsu binarized image
- The intensity of the image that filtered out the central part based on Otsu or Multi-Region Binarization as shown in Figure 5.5

FIGURE 5.5: Flow chart of the procedures to filter out the central part of the greyscale image

The feature sets extracted from greyscale images were listed in Table 5.3, and the boxplot check results are summarized in Table 5.4. Since the features from greyscale images of intensity type 2, 3, 4, 5, 6 did not pass the boxplot check, there was no feature in G_2 , G_3 , G_4 , G_5 , G_6 . Thus, G_2 , G_3 , G_4 , G_5 , G_6 were not used in later classification experiments.

TABLE 5.3: Features extracted from greyscale images

EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients.

Intensity type	EHHOG	Features passed the boxplot check
	Extracted from	
1	Original greyscale images	G1
2	(2-Region – Otsu) filtered images	G2
3	(3-Region – Otsu) filtered images	G3
4	(4-Region – Otsu) filtered images	G4
5	(5-Region – Otsu) filtered images	G5
6	(6-Region – Otsu) filtered images	G6
7	Centre filtered images based on Otsu binarization	G7
8	Centre filtered images based on 2-Region binarization	G8
9	Centre filtered images based on 3-Region binarization	G9
10	Centre filtered images based on 4-Region binarization	G10
11	Centre filtered images based on 5-Region binarization	G11
12	Centre filtered images based on 6-Region binarization	G12

TABLE 5.4: Boxplot check results of features extracted from greyscale images on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result
	EHHOG
1	1
2	0
3	0
4	0
5	0
6	0
7	1
8	1
9	1
10	1
11	1
12	1

5.3.3 Feature Combination Used in Classification Experiments

A basic concept of feature selection (i.e., combining which features together, excluding which features) for classification is that the features should be strongly correlated to the target (i.e., be able to differentiate the classes) and the features should not be too similar among each other because redundancy can harm the classification accuracy, indicating that more features do not mean higher accuracy [57]–[60].

Based on this concept, 1 feature set from the greyscale images, 1 feature set from Otsu binarized images, and 1 feature set from Multi-Region binarized images were combined to classify our images. The feature combination was referred to as $G\beta B1B\alpha$ (details shown in (5.2)). From Tables 5.3, 5.4, there was no feature in $G2$, $G3$, $G4$, $G5$, $G6$, so only $\beta = 1, 7, 8, 9, 10, 11, 12$ were used in experiments.

$$G\beta B1B\alpha$$

$$\text{where } \beta = 1, \alpha = 2, 3, \dots, 6$$

$$\beta = 7, \alpha = 2, 3, \dots, 6$$

$$\beta = 8, \alpha = 2, 3, \dots, 6$$

(5.2)

$$\beta = 9, \alpha = 2, 3, \dots, 6$$

$$\beta = 10, \alpha = 2, 3, \dots, 6$$

$$\beta = 11, \alpha = 2, 3, \dots, 6$$

$$\beta = 12, \alpha = 2, 3, \dots, 6$$

5.4 Classification Results and Discussions

5.4.1 Different Feature Combinations

The classification results of feature combination $G\beta B1B\alpha$ described in Section 5.3.3 are presented in Table 5.5.

From Table 4.8, the EHHOG method using EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) from original greyscale images and aspect ratio (AR), solidity (SO) extracted from Otsu binarized images (i.e., the feature combination $G1B1$ by the definition in Section 5.3) with radial basis function (RBF) support vector machine (SVM), the accuracy was 91.7%, and with bagging decision trees (BDT), the accuracy was 92.2%. The existing Bueno's method [49] with linear SVM and the deep-learning-based Pedraza's method [73] had the accuracy 93.0% and 93.9% respectively but requires high computational cost. In Table 5.5, the highest testing accuracy was 93.4% with the feature combination $G1B1B3$ or $G1B1B4$, which is a good improvement from the EHHOG method. Also, the numbers of features were 5 for $G1B1B3$ and 4 for $G1B1B4$ (see Tables 5.2, 5.4), very close to 3 for the EHHOG method. With similar feature extraction procedures, the computational complexity (43 MFLOPs (million floating-point operations) for $G1B1B3$ to test one image) was similarly low as the EHHOG method (41 MFLOPs).

Using the feature combination $G1B1B2B3B4B5B6$ with RBF SVM, the testing accuracy was 92.2%, lower than the best case's testing accuracy 93.4% of using $G1B1B\alpha$, confirming

the feature selection concept that more features do not mean higher accuracy so the features should be combined selectively for classification (Section 5.3.3).

The feature combinations with the features extracted from filtered intensities ($G7, G8, \dots, G12$) had much lower testing accuracy compared to $G1B1B2$, $G1B1B3$, $G1B1B4$, or $G1B1B5$. For further analysis, the training accuracies are also listed in Table 5.5. It can be observed that all the training accuracies were higher than 90%, but for the feature combinations with the features extracted from filtered intensities ($G7, G8, \dots, G12$), the differences between the training accuracy and the testing accuracy were very high (more than 20%, even up to 40% difference), showing that the classifier was over-fitted. Over-fitting may be caused by different reasons, like insufficient samples and training time. The number of samples, number of features, and training procedures were similar for the feature combinations with $G1$ (testing accuracies all higher than 90%) and with $G7, G8, \dots, G12$ (testing accuracies between 45% to 75%), so a possible reason is that the features in $G7, G8, \dots, G12$ were not as good the feature in $G1$ in terms of classification ability. This indicates that although human eyes cannot tell the differences of the image background (the outermost dark part of the image) among different classes (Section 5.3.2), the computer calculated feature EHHOG from the whole original greyscale image contains essential information for classification that missed by EHHOG from the central part intensities filtered from the original image. The feature EHHOG summarized whether an image has more "non-edge" cells or more "edge" cells (Section 4.3.2). Filtering central intensities would make the "scattering pattern" part the same size as the filter, and thus lost the differences in the ratio of the "scattering pattern" part and the "background" part for different classes.

TABLE 5.5: Classification results of feature combination $G\beta B1B\alpha$ with SVM on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. SVM: support vector machine. SVMs used here were with RBF (radial basis function) kernels. With 10-fold cross-validation, the accuracies displayed are the mean value of the 10 folds.

Feature combination	Training accuracy	Testing accuracy
$G1B1B2$	97.4%	91.7%
$G1B1B3$	97.6%	93.4%
$G1B1B4$	96.4%	93.4%

G1B1B5	96.8%	92.2%
G1B1B6	96.9%	92.2%
G7B1B2	94.6%	74.3%
G7B1B3	95.8%	74.8%
G7B1B4	93.3%	73.6%
G7B1B5	97.0%	74.5%
G7B1B6	94.3%	71.6%
G8B1B2	95.8%	56.6%
G8B1B3	97.3%	57.4%
G8B1B4	94.9%	56.4%
G8B1B5	97.2%	59.6%
G8B1B6	95.1%	56.4%
G9B1B2	96.5%	48.5%
G9B1B3	97.1%	52.9%
G9B1B4	95.3%	50.2%
G9B1B5	97.0%	51.4%
G9B1B6	95.6%	50.7%
G10B1B2	96.0%	54.6%
G10B1B3	95.3%	56.0%
G10B1B4	94.4%	54.3%
G10B1B5	96.7%	56.0%
G10B1B6	94.4%	54.3%
G11B1B2	95.4%	48.8%
G11B1B3	93.7%	51.8%
G11B1B4	92.5%	53.5%
G11B1B5	94.7%	54.4%
G11B1B6	93.5%	52.7%
G12B1B2	94.7%	49.7%
G12B1B3	93.4%	52.2%
G12B1B4	90.0%	51.2%
G12B1B5	93.5%	54.3%

G12B1B6	91.3%	50.5%
---------	-------	-------

The classification results of using features extracted from greyscale images and binarized images together and separately are presented in Table 5.6. Combining features from Otsu binarized images and Multi-Region binarized images ($B1B3$, $B1B2$) gives higher testing accuracy than using the features separately ($B1$, $B2$, $B3$), proving that the proposed Multi-Region Binarization method can improve the classification performance on our waterborne pathogen images. Adding the proposed feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) from original greyscale images ($G1B1B3$) further improved the testing accuracy. Similar trends existed for $G1B1B2$, $G1B1B4$, and $G1B1B5$. However, if EHHOG from central part filtered images were used, the testing accuracies of both $G7$ and $G7B1B2$ were quite low, even lower than only using $B1B2$. Similar trends existed for other feature combinations with $G8$, $G9$, $G10$, $G11$, and $G12$. This is consistent with the previous discussions for Table 5.5 that more features do not mean higher accuracy and that EHHOG from the whole original greyscale image contains essential information for classification that missed by EHHOG from the filtered intensities. Thus, $G7, G8, \dots, G12$ would not be used to develop the classification algorithm. The algorithm using the proposed Multi-Region Binarization and the feature combination $G1B1B\alpha$ (training and testing all $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$) and choosing the one with the highest testing accuracy) was proposed to improve the previous EHHOG algorithm. It is named Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM).

TABLE 5.6: Examples of classification results comparing using features extracted from greyscale and binarized images together and separately with SVM on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. SVM: support vector machine. SVMs used here were with RBF (radial basis function) kernels. Examples shown here are feature combinations *G1B1B3* and *G7B1B2*. With 10-fold cross-validation, the accuracies displayed are the mean value of the 10 folds.

Feature	Training accuracy	Testing accuracy
G1	72.6%	74.9%
B1	82.1%	77.8%
B3	72.8%	73.2%
B1B3	89.6%	84.6%
G1B1B3	97.6%	93.4%
G7	68.9%	48.0%
B1	82.1%	77.8%
B2	81.0%	78.1%
B1B2	88.7%	83.5%
G7B1B2	94.6%	74.3%

The accuracies with radial basis function (RBF) support vector machine (SVM) and bagging decision trees (BDT) were similar, so only the results of RBF SVM were presented here for the investigation of different feature combinations.

5.4.2 Other Local Thresholding Binarization Methods

A local thresholding binarization method, Multi-Region Binarization, was proposed and both Otsu (a global thresholding binarization method) [24] and Multi-Region binarized images were used in classification as discussed in Sections 5.3, 5.4.1. For comparison, 2 existing local thresholding methods, Sauvola's method [101] and K-means based method [102], were used.

Binarized Image Comparison

The binarized images with different binarization methods of some image examples are compared in Figure 5.6. Since the details are not clearly shown in Figure 5.6, the individual

images were shown in Figures 5.7, 5.8, 5.9, 5.10, 5.11, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.18, 5.19, 5.20, and 5.21.

Comparing the original image and binarized images, it can be observed that the proposed Multi-Region Binarization method keeps more outer ring-like/ irregular patterns than Otsu's method, Sauvola's method, and K-means based method. In Figure 5.16, there are some unnatural horizontal lines because the Multi-Region Binarization partitions the regions based on rectangles. But generally, the Multi-Region binarization images segment out both the circle-/ellipse-like inner patterns and the ring-like irregular outer patterns relatively well. The images binarized by Sauvola's method keep fewer outer patterns than Otsu's method. The K-means based method has a similar object capture range as Otsu's method but with jagged boundaries.

FIGURE 5.6: Image example of original images and binarized images with different binarization methods

Otsu's method [24] is a global thresholding binarization method. Sauvola's method [101] and K-means based method [102] are 2 local thresholding binarization methods. Multi-Region method is the proposed local thresholding binarization method. Multi-Region Binarization illustrated here used 3 regions.

FIGURE 5.7: Image example: original image of class G, #120 image

FIGURE 5.8: Image example: binarized image with Otsu's method [24] of class G, #120 image

FIGURE 5.9: Image example: binarized image with Sauvola's method [101] of class G, #120 image

FIGURE 5.10: Image example: binarized image with K-means based method [102] of class G, #120 image

FIGURE 5.11: Image example: binarized image with proposed Multi-Region method of class G, #120 image

FIGURE 5.12: Image example: original image of class C, #107 image

FIGURE 5.13: Image example: binarized image with Otsu's method [24] of class C, #107 image

FIGURE 5.14: Image example: binarized image with Sauvola's method [101] of class C, #107 image

FIGURE 5.15: Image example: binarized image with K-means based method [102] of class C, #107 image

FIGURE 5.16: Image example: binarized image with proposed Multi-Region method of class C, #107 image

FIGURE 5.17: Image example: original image of class O, #15 image

FIGURE 5.18: Image example: binarized image with Otsu's method [24] of class O, #15 image

FIGURE 5.19: Image example: binarized image with Sauvola's method [101] of class O, #15 image

FIGURE 5.20: Image example: binarized image with K-means based method [102] of class O, #15 image

FIGURE 5.21: Image example: binarized image with proposed Multi-Region method of class O, #15 image

Classification Result Comparison

The results of the feature combination $G1B1B\alpha$ comparing the proposed Multi-Region Binarization and the 2 existing local thresholding binarization methods are shown in Table 5.7 below. Here, $G1$ (the features extracted from the original greyscale images and passed the boxplot check), $B1$ (the features extracted from the global thresholding (Otsu) binarized images and passed the boxplot check) are the same as in Section 5.3, and $B\alpha$ refers to the features extracted from local thresholding binarized images and could pass the boxplot check.

The testing accuracies using the proposed Multi-Region Binarization were always higher than using the 2 existing binarization methods [101],[102] with either radial basis function (RBF) support vector machine (SVM) or bagging decision trees (BDT) as the classifier, indicating that the shape of the outer patterns on our images also contains information that is useful for classification.

From Tables 4.8, 5.7, the accuracies with RBF SVM and BDT were similar, so both classifiers will be used in later experiments to select the one with the better testing accuracy.

TABLE 5.7: Comparison of proposed Multi-Region Binarization and existing local thresholding binarization methods in classification using feature combination $G1B1B\alpha$ on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. SVM: support vector machine. BDT: bagging decision trees. SVMs used here were with RBF (radial basis function) kernels. * denotes the proposed method. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Binarization method for $B\alpha$	Testing accuracy	
	SVM	BDT
Multi-Region Binarization *	93.4%	92.4%
Sauvola's method [101]	89.6%	87.4%
K-means based method [102]	88.6%	88.8%

5.5 Summary of Proposed BiGHAM Algorithm

Chapter 4 presented the proposed EHHOG algorithm using morphological features aspect ratio (AR) and solidity (SO) extracted from Otsu binarized images and a proposed features EHHOG (entropy of HOG) (HOG: histogram of oriented gradients) together to classify our waterborne pathogen images. In this Chapter, a new algorithm was presented to improve the classification accuracy while keeping a low computational cost for the aim of future portable, real-time water monitoring applications. A local thresholding binarization method, Multi-Region Binarization, was proposed according to the appearances of our images to catch the information missed by the global thresholding Otsu binarization. The scattering patterns of our images can be regarded as 2 parts: the inner brighter circle-/ellipse-like part, and the outer dimmer ring-like/irregular part. Otsu binarization only kept the inner part, while the proposed Multi-Region Binarization can segment out both parts. With Multi-Region Binarization, the morphological feature compactness (CO) could pass the boxplot check in some cases, so AR, SO, CO were all extracted and used as the features from binarized images. (Section 5.2)

Based on the concept in the image classification field that the features should be strongly correlated to the target and weakly correlated among each other, 1 feature set extracted from greyscale images, 1 feature set extracted from Otsu binarized images, and 1 feature set extracted from Multi-Region binarized images (the number of regions is 2, 3, 4, 5, or 6) were combined for classification. Only features that could pass the boxplot check were used in

the classification experiments. The feature EHHOG extracted from the original intensities, the (Multi-Region Binarization – Otsu binarization) filtered intensities, and the central part filtered intensities were tried but only EHHOG from the original greyscale images turned out useful in the classification experiments. The feature combination used for classification can be referred to as $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). The proposed classification algorithm using Multi-Region Binarization and the feature combination $G1B1B\alpha$ was called Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM). (Section 5.3)

Among all $G1B1B\alpha$ combinations, $G1B1B3$ and $G1B1B4$ reached 93.4% with radial basis function (RBF) support vector machine (SVM) as the classifier, which is a good improvement from 92.2% of EHHOG algorithm with bagging decision trees (BDT) with a similar computational complexity (EHHOG algorithm: 41 MFLOPs (floating-point operations), BiGHAM: 43 MFLOPs). The accuracy 93.4% with the proposed BiGHAM method was higher than 93.0% using the existing handcrafted-feature-based microorganism classification algorithm, Bueno's method [49] with linear SVM (61 MFLOPs) at a lower computational cost. The deep-learning-based Pedraza's method [73] has a higher accuracy of 93.9% but requires a much higher computational cost (716 MFLOPs), and thus not suitable for portable, real-time applications. Additionally, using 2 existing local thresholding binarization methods [101],[102] instead of the proposed Multi-Region Binarization in the feature combination $G1B1B\alpha$, the classification accuracies were less than 90%, proving the usefulness of the proposed Multi-Region Binarization method. (Section 5.4)

Conceptually, the Multi-Region Binarization method and BiGHAM algorithm can be extended to other images that have scattering patterns or appear brighter in the inner part of the image (to be discussed in Chapter 7). Based on EHHOG and BiGHAM algorithms, Chapter 6 continues to improve the algorithm aiming at increasing the classification accuracy while keeping the algorithm simple.

Chapter 6

Mutual-Information-Based Feature Selection to Improve Waterborne Pathogen Classification

6.1 Introduction

Chapter 5 presented the proposed method, Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM), using the proposed Multi-Region Binarization method and the feature combination $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$) for classification. $G1$ refers to the feature set extracted from the original greyscale images, $B1$ refers to the feature set extracted from Otsu binarized images, and $G\alpha$ refers to the feature set extracted from Multi-Region binarized images. For all $G1B1B\alpha$ combinations, the classifier needed to be trained and tested 5 times to choose the best feature combination. The region partitioning for Multi-Region Binarization was based on rectangles, which sometimes made unnatural horizontal or vertical lines in the binarized images (Figure 5.16). Thus, in this Chapter, feature selection methods were discussed and a new way to apply the mutual-information-based feature ranking was proposed. The feature combination $G1B1ARSOCO$ after the feature selection with a new circle-based region partition method for Multi-Region Binarization achieved better accuracies than BiGHAM method and only required 1 time of classifier training and testing.

6.2 Mutual-Information-Based Feature Selection

As discussed in Sections 5.3.3, 5.4.1, adding more features does not certainly give a classification improvement. Sometimes a subset can achieve higher testing accuracy than using all the features, so feature selection (FS) can help to improve the classification accuracy. The principle of FS for classification is to choose features that are highly correlated with the target classes but not similar among each other [57]–[60].

6.2.1 Feature Ranking Criterion

The minimal-Redundancy-Maximal-Relevance (mRMR) criterion proposed by H. Peng, et al. [25] was applied for the classification of our optofluidic images. The mRMR method is a widely used feature selection (FS) method until today [61]–[64]. Redundancy refers to the dependency among the features. If one feature is highly correlated with other feature(s), then it is redundant. Relevance refers to the dependency between the feature and the target. If one feature does not have a strong correlation with the targets, then it is irrelevant.

The dependency of two variables is measured by the mutual information I as in Equation (6.1), so mRMR is a mutual-information-based feature selection method.

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \quad (6.1)$$

Here, $p(x)$, $p(y)$ denote the marginal probabilities of random variables X and Y respectively, $p(x,y)$ denotes the joint probability of X and Y .

The mRMR criterion is to maximize Equation (6.2).

$$relevance - redundancy = I(x_j; c) - \frac{1}{m-1} \sum_{x_i \in S_{m-1}} I(x_j; x_i) \quad (6.2)$$

Here, I denotes the mutual information, S_{m-1} denotes the selected subset, x_j denotes the feature under consideration in the set $X - S_{m-1}$, c denotes the target, X denotes the original feature set (i.e., all the extracted features before FS). There are $m-1$ features in the set S_{m-1} .

The mRMR method ranks the features as the following procedures.

- 1) At the beginning, the selected subset S_{m-1} has no feature, and all features are in the original feature set X . The first feature was selected by maximizing $I(x_j; c)$.

- 2) The selected feature was moved from X to S_{m-1} . The m th feature was selected from the set $X - S_{m-1}$ by maximizing Equation (6.2).
- 3) Step 2 is iterated until there is no feature left in $X - S_{m-1}$.
- 4) The first selected feature is the rank 1 feature. The second selected feature is the rank 2 feature. ... The m th selected feature was the rank m feature.

6.2.2 Ways to Apply Feature Selection

In existing mRMR applications, incremental feature selection (IFS) is usually used. All the features are ranked by mRMR criterion, and N subsets are formed by adding 1 feature each time (called IFS feature sets later): 1-D feature set has the rank 1 feature, 2-D feature set has the rank 1 and the rank 2 feature, ..., N -D feature set with all the features (N denotes the number of features in the original feature set). All the N feature sets should be trained and tested to determine which one has the best classification performance.

Feature Ranking on All Features

In our previous study, features extracted from original greyscale images (feature set $G1$), Otsu binarized images (feature set $B1$), and Multi-Region binarized images (feature set $B2, B3, B4, B5, B6$) were used to classify our images (Tables 5.1, 5.3, Section 5.4.1). A direct application of minimal-Redundancy-Maximal-Relevance (mRMR) feature selection (FS) is to rank all the features in $G1B1B2B3B4B5B6$ and classify the images with all IFS feature sets.

However, this method did not improve the testing accuracy from the proposed Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) method and even required more times of training and testing. BiGHAM method required to train and test for 5 times for all $G1B1B\alpha$ feature combinations, while applying mRMR directly on $G1B1B2B3B4B5B6$, which has 10 or 12 features according to Tables 6.1, 6.2, required to train and test for 10 or 12 times. The experimental results are presented in Section 6.4.1.

Feature Ranking on Each Morphological Feature Separately

As discussed in Chapter 5, the proposed feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) from the original greyscale images ($G1$) always increased the accuracy when combining with features from binarized images, because it has

quite different value ranges from the morphological features of the binarized images on the boxplot. For binarized images, Otsu binarized images (generating $B1$) captured only the inner part of the scattering pattern, while Multi-Region binarized images (generating $B\alpha$ ($\alpha = 2, 3, 4, 5, 6$) with 2, 3, 4, 5, 6 regions) captured both the inner and outer parts, and the images binarized by Multi-Region method with different numbers of regions were similar compared to the difference between images binarized by Otsu's method and Multi-Region Binarization.

Based on this prior knowledge, a new way to implement the mRMR FS on classification was proposed as the procedures below. Each morphological feature (aspect ratio (AR), solidity (SO), compactness (CO)) from the Multi-Region binarized images was ranked separately, and the rank 1 AR, SO, CO were selected accordingly.

- 1) The features in $G1$ and $B1$ are put into the selected subset S_{m-1} directly.
- 2) One AR is selected from all the ARs in $B\alpha$ ($\alpha = 2, 3, 4, 5, 6$) by maximizing Equation (6.2). Here, the feature set under consideration $X - S_{m-1}$ is all the ARs extracted from Multi-Region binarized images.
- 3) One SO is selected from all the SOs in $B\alpha$ ($\alpha = 2, 3, 4, 5, 6$) by maximizing Equation (6.2). Here, the feature set under consideration $X - S_{m-1}$ is all the SOs extracted from Multi-Region binarized images.
- 4) One CO is selected from all the COs in $B\alpha$ ($\alpha = 2, 3, 4, 5, 6$) by maximizing Equation (6.2). Here, the feature set under consideration $X - S_{m-1}$ is all the COs extracted from Multi-Region binarized images.
- 5) The feature set combining $G1$, $B1$, and the selected AR, SO, CO is named $G1B1ARSOCO$.

In this way, only 1 time of training and testing is required on the feature combination $G1B1ARSOCO$.

6.3 Rectangle-Based and Circle-Based Multi-Region Binarization

In Section 5.2.1, the Multi-Region Binarization method was proposed for binarization with local thresholds. The regions were partitioned based on rectangles as shown in Figure 5.1. Alternatively, the regions can be partitioned based on circles (Figure 6.1) as well.

FIGURE 6.1: Illustration of circle-based Multi-Region Binarization (number of regions = 3)

6.3.1 Comparison of Binarized Images

Figures 6.2, 6.3 shows image examples of an original image, the binarized image by Otsu's method [24], and rectangle-based/circle-based Multi-Region Binarization method. It can be observed that Otsu binarization only kept the inner brighter circle-like pattern while Multi-Region Binarization also kept the outer dimmer ring-like pattern, which is consistent with the idea of proposing Multi-Region Binarization to capture the missed information as discussed in Section 5.2.2, 5.4.2. For the number of regions of the Multi-Region Binarization method, the images binarized with 7 regions upwards were similar to the 6-region method, so 2, 3, 4, 5, 6 regions were used in the classification experiments.

For clearer viewing, the Multi-Region binarized images that have obvious differences between rectangle-based and circle-based region partition methods were enlarged and shown in Figures 6.4, 6.5, 6.6, and 6.7 individually. Comparing Figure 6.2 and Figure 6.3, the Multi-Region binarized images with rectangle-based and circle-based methods seem similar at most times.

FIGURE 6.2: Image example of our dataset: original, Otsu binarized, rectangle-based Multi-Region (number of regions = 2, 3, 4, 5) binarized images

FIGURE 6.3: Image example of our dataset: original, Otsu binarized, circle-based Multi-Region (number of regions = 2, 3, 4, 5) binarized images (Multi-Region Binarized images displayed are the rectangle-based)

Figure 6.4 and Figure 6.5 show that sometimes the circle-based Multi-Region Binarization kept a more complete shape of the scattering pattern than the rectangle-based method, but this only happened few times among all the images. In Figure 6.6, there exists an unnatural horizontal line caused by the rectangle-based region partitioning similar to Figure 5.16, while the Multi-Region Binarization with circle-based regions does not have this problem (Figure 6.7). From the definitions of the features aspect ratio (AR), solidity (SO), and compactness (CO) (see Table 4.1, Appendix F), the AR and SO for Figure 6.4 and Figure 6.5 would be similar, but the CO values would be different.

FIGURE 6.4: Image example: rectangle-based 3-Region binarized images of class C, #74 image

FIGURE 6.5: Image example: circle-based 3-Region binarized images of class C, #74 image

FIGURE 6.6: Image example: rectangle-based 5-Region binarized images of class C, #74 image

FIGURE 6.7: Image example: circle-based 5-Region binarized images of class C, #74 image

In summary, the 2 region partition methods give similar results, but sometimes the images binarized with circle-based Multi-Region Binarization can capture the scattering patterns smoother and more complete. As for the classification performance, experiments need to be implemented to determine the effect of using different region partition methods.

6.3.2 Comparison of Boxplot Check Results

The boxplot check results of using rectangle-based and circle-based Multi-Region Binarization methods are displayed in Tables 6.1, 6.2. The boxplot check results were the same, except for the compactness (CO) passing the boxplot check for 5-region and 6-region cases with circle-based Multi-Region Binarization. This is consistent with the analysis of binarized images in Section 6.3.1.

TABLE 6.1: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. AR: aspect ratio. SO: solidity. CO: compactness. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check. This table is a combination of Table 5.1 and Table 5.2

Binarization method	Boxplot check result		
	AR	SO	CO
Otsu's method [24]	1	1	0
Multi-Region Binarization, 2 regions	1	1	0
Multi-Region Binarization, 3 regions	1	1	0
Multi-Region Binarization, 4 regions	1	0	0
Multi-Region Binarization, 5 regions	1	0	0
Multi-Region Binarization, 6 regions	1	0	0

TABLE 6.2: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. AR: aspect ratio. SO: solidity. CO: compactness. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result		
	AR	SO	CO
Otsu's method [24]	1	1	0
Multi-Region Binarization, 2 regions	1	1	0
Multi-Region Binarization, 3 regions	1	1	0
Multi-Region Binarization, 4 regions	1	0	0
Multi-Region Binarization, 5 regions	1	0	1
Multi-Region Binarization, 6 regions	1	0	1

6.4 Classification Results and Discussions

6.4.1 Experiments on Our Original Dataset

mRMR Ranking on All Features

According to the boxplot check result in Table 6.1, the features in the feature set $G1B1B2B3B4B5B6$ with rectangle-based Multi-Region Binarization are listed in Table 6.3. The incremental feature selection (IFS) feature sets based on the ranking by applying the minimal-Redundancy-Maximal-Relevance (mRMR) criterion on $G1B1B2B3B4B5B6$ (procedures explained in Section 6.2) are shown in Table 6.4.

TABLE 6.3: Features in the feature set $G1B1B2B3B4B5B6$ with rectangle-based Multi-Region Binarization

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. CO: compactness.

Number	Feature	Short name	From feature set
1	EHHOG of original greyscale images	G1_EHHOG	$G1$
2	AR of Otsu binarized images	B1_AR	$B1$
3	SO of Otsu binarized images	B1_SO	
4	AR of 2-Region binarized images	B2_AR	$B2$
5	SO of 2-Region binarized images	B2_SO	
6	AR of 3-Region binarized images	B3_AR	$B3$
7	SO of 3-Region binarized images	B3_SO	
8	AR of 4-Region binarized images	B4_AR	$B4$
9	AR of 5-Region binarized images	B5_AR	$B5$
10	AR of 6-Region binarized images	B6_AR	$B6$

TABLE 6.4: Features in the IFS feature sets of mRMR ranking on $G1B1B2B3B4B5B6$ with rectangle-based Multi-Region Binarization

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. IFS: incremental feature selection. mRMR: minimal-Redundancy-Maximal-Relevance.

IFS feature set	Features as in Table 6.3
1	1
2	1, 2
3	1, 2, 5
4	1, 2, 5, 6
5	1, 2, 5, 6, 3
6	1, 2, 5, 6, 3, 4
7	1, 2, 5, 6, 3, 4, 7
8	1, 2, 5, 6, 3, 4, 7, 8
9	1, 2, 5, 6, 3, 4, 7, 8, 10
10	1, 2, 5, 6, 3, 4, 7, 8, 10, 9

The classification results on the incremental feature selection (IFS) feature sets that were formed by directly applying the minimal-Redundancy-Maximal-Relevance (mRMR) ranking on all the features in $G1B1B2B3B4B5B6$ were presented in Table 6.5. These results proved again that the testing accuracy would not increase monotonically with the increase of the number of features as discussed in Sections 5.3.3, 5.4.1, 6.2.

In Table 5.7, the testing accuracies of the proposed Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) method using feature combination $G1B1B\alpha$ on our dataset were 93.4% and 92.4% with radial basis function (RBF) support vector machine (SVM) and bagging decision trees (bagging decision trees) as the classifier. In Table 6.5, the highest accuracies with RBF SVM and BDT were not improving. Besides, the method of Table 6.5 was more complex for requiring 10 times of training and testing to find the best feature set while the BiGHAM method only required 5 times. Thus, a direct application of the mRMR ranking on all the features was not recommended for classification on our images.

TABLE 6.5: Classification results of applying mRMR ranking on $G1B1B2B3B4B5B6$ with RBF SVM and BDT on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. mRMR: minimal-Redundancy-Maximal-Relevance. RBF: radial basis function. SVM: support vector machine. BDT: bagging decision trees. IFS: incremental feature selection. IFS feature sets are listed in Table 6.4. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

IFS feature set	Number of features	Testing accuracy	
		SVM	BDT
1	1	74.9%	63.2%
2	2	79.8%	73.9%
3	3	90.5%	91.0%
4	4	90.5%	89.0%
5	5	91.7%	92.2%
6	6	91.2%	91.5%
7	7	93.4%	92.2%
8	8	92.7%	92.2%
9	9	92.9%	92.4%
10	10	92.2%	90.7%

Feature Combination $G1B1B\alpha$ and $G1B1ARSOCO$ with with Rectangle-Based and Circle-Based Regions

A new way to apply minimal-Redundancy-Maximal-Relevance (mRMR) ranking was proposed in Section 6.2.2. This mutual-information-based feature selection (FS) based on mRMR ranking gives a feature combination $G1B1ARSOCO$. The classification results using the proposed feature combination $G1B1B\alpha$ (details presented in Chapter 5) and $G1B1ARSOCO$ with rectangle-based and circle-based Multi-Region Binarization methods are summarized in Tables 6.6, 6.7.

The number of features (5 for the best $G1B1B\alpha$, 6 for $G1B1ARSOCO$) and feature extraction procedures of $G1B1B\alpha$ and $G1B1ARSOCO$ were quite close, so the computational complexities of testing 1 image were similarly low for the 2 feature combinations. Changing $G1B1B\alpha$ to $G1B1ARSOCO$ (when region partition method unchanged) or changing rectangle-based region partitioning to circle-based (when feature combination unchanged) did not always improve the classification accuracy, but using $G1B1ARSOCO$ with circle-based regions always had higher testing accuracy than using $G1B1B\alpha$ with rectangle-based regions. Additionally, it required 5 times of training and testing to find out the best feature set among all $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$), but only 1 time for the feature combination $G1B1ARSOCO$. In Chapter 5, using the feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization was named Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM). Here, using the feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization was named improved-BiGHAM (iBiGHAM) for the improvement on both classification accuracy and algorithm simplicity.

TABLE 6.6: Classification results of $G1B1B\alpha$ and $G1B1ARSOCO$ with RBF SVM on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. RBF: radial basis function. SVM: support vector machine. The testing accuracies of the feature combination $G1B1B\alpha$ displayed in this table are the best results among all $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Regions for Multi-Region Binarization	Testing accuracy	
	$G1B1B\alpha$	$G1B1ARSOCO$
Rectangle-based	93.4%	95.9%
Circle-based	94.9%	93.9%

TABLE 6.7: Classification results of $G1B1B\alpha$ and $G1B1ARSOCO$ with BDT on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. BDT: bagging decision trees. The testing accuracies of the feature combination $G1B1B\alpha$ displayed in this table are the best results among all $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Regions for Multi-Region Binarization	Testing accuracy	
	$G1B1B\alpha$	$G1B1ARSOCO$
Rectangle-based	92.4%	92.4%
Circle-based	92.0%	93.2%

6.4.2 Experiments on Our Dataset with Different Sample Numbers

To investigate the effect of the dataset size (how many samples/ images per class), data augmentation was implemented on our images. Two-degree rotations were used on each image to generate more than 1000 images per class, and then 30, 70, 100, 300, 700, 1000 image-per-class datasets were randomly selected (image examples illustrated in Appendix H). Since the boundaries and outer parts of our images were black or close to black, image rotation was padded with black (intensity = 0).

Proposed EHHOG Method

The classification results of applying the proposed EHHOG method of combining features extracted from the binarized images (aspect ratio (AR), solidity (SO)) and from the greyscale

images (entropy of histogram of histogram of oriented gradients (EHHOG)) on these augmented datasets were put in Table 6.8 below.

TABLE 6.8: Classification results of EHHOG method on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. SVM: support vector machine. BDT: bagging decision trees. SVMs used here were with RBF (radial basis function) kernels. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Number of images per class	Testing accuracy	
	SVM	BDT
30	76.7%	82.2%
70	86.2%	88.1%
100	89.7%	91.7%
300	91.0%	93.8%
700	91.7%	97.0%
1000	93.0%	97.7%

With the increase of the number of samples per class, the testing accuracy also increases (Table 6.8). The accuracies using bagging decision trees (BDT) were higher than those using support vector machine (SVM) as the classifier. The trend was the same as for the dataset without augmentation in the last row of Table 4.6.

Proposed Feature Combination $G1B1B\alpha$

The classification results of using the proposed feature combination $G1B1B\alpha$ (training and testing for 5 times to find the best feature combination among all $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$)) with rectangle-based and circle-based Multi-Region Binarization were put in Tables 6.9, 6.10 below.

TABLE 6.9: Classification results of feature combination $G1B1B\alpha$ with RBF SVM on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. Num. refers to the number of images per class. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Num.	Multi-Region Binarization			
	Rectangle-based		Circle-based	
	Accuracy	From feature combination	Accuracy	From feature combination
30	77.8%	G1B1B3	74.4%	G1B1B3
70	86.7%	G1B1B2	87.6%	G1B1B2
100	92.3%	G1B1B3	94.3%	G1B1B6
300	94.0%	G1B1B2	94.6%	G1B1B2
700	94.3%	G1B1B2	94.8%	G1B1B2
1000	94.8%	G1B1B2	95.1%	G1B1B2

TABLE 6.10: Classification results of feature combination $G1B1B\alpha$ with BDT on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. BDT: bagging decision trees. Num. refers to the number of images per class. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Num.	Multi-Region Binarization			
	Rectangle-based		Circle-based	
	Accuracy	From feature combination	Accuracy	From feature combination
30	82.2%	G1B1B2	83.3%	G1B1B2
70	90.5%	G1B1B2	88.6%	G1B1B2
100	91.3%	G1B1B2	90.3%	G1B1B2
300	95.8%	G1B1B2	95.7%	G1B1B2
700	98.3%	G1B1B2	98.4%	G1B1B2
1000	98.7%	G1B1B2	99.1%	G1B1B2

Similar to Table 6.8, the testing accuracy increased as the number of image samples increased. Generally, the testing accuracies with bagging decision trees (BDT) as the classifier

were higher than those with radial basis function (RBF) support vector machine (SVM), except for the 100-image-per-class case. In Table 5.7, the testing accuracy of using feature combination $G1B1B\alpha$ on our original dataset (without augmentation) with RBF SVM (93.4%) was higher than using BDT (92.4%) as the classifier. Thus, both classifiers were used in the classification experiments as discussed in Section 5.4.2. In most of the time, the accuracy with circle-based Multi-Region Binarization was higher than that with rectangle-based method, but the difference was not very large. Comparing Table 6.8, and Tables 6.9, 6.10, the testing accuracies using feature combination $G1B1B\alpha$ were higher than using EHHOG method in most of the cases as discussed in Section 5.4.1, confirming that proposing the Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) was able to improve classification compared to the proposed EHHOG method.

Proposed Feature Combination $G1B1ARSOCO$

The classification results of using the proposed feature combination $G1B1ARSOCO$ (proposed in Section 6.2.2) with rectangle-based and circle-based Multi-Region Binarization were put in Tables 6.11, 6.12 below.

TABLE 6.11: Classification results of feature combination $G1B1ARSOCO$ with RBF SVM on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. Num. refers to the number of images per class. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Num.	Multi-Region Binarization	
	Rectangle-based	Circle-based
30	83.3%	85.6%
70	87.6%	87.6%
100	88.7%	92.0%
300	93.2%	95.8%
700	96.3%	97.1%
1000	94.1%	96.6%

TABLE 6.12: Classification results of feature combination *G1B1ARSOCO* with BDT on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. BDT: bagging decision trees. Num. refers to the number of images per class. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Num.	Multi-Region Binarization	
	Rectangle-based	Circle-based
30	83.3%	81.1%
70	89.5%	90.0%
100	91.7%	95.3%
300	95.6%	97.0%
700	98.0%	98.6%
1000	98.5%	99.2%

To compare the performances of using the feature combination *G1B1B α* and *G1B1ARSOCO*, rectangle-based and circle-based regions with different numbers of samples, the testing accuracies with the radial basis function (RBF) support vector machine (SVM) as the classifier (Table 6.9, Table 6.11) was plotted in Figure 6.8; the testing accuracies with the bagging decision trees (BDT) as the classifier (Table 6.10, Table 6.12) was plotted in Figure 6.9. The classification results of the proposed EHHOG method (Chapter 4), Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) method (Chapter 5), and improved-BiGHAM (iBiGHAM) method (Chapter 6) on our augmented datasets were plotted in Figure 6.10.

In Figures 6.8, 6.9, and 6.10, the exhaustive search (ES) on all the features in *G1B1B2B3B4B5B6* (*G1B1toB6* for short) is also plotted as the reference. ES is to train and test with all the subsets in *G1B1toB6* to get the highest possible testing accuracy. From the theory of mathematical combination, the number of subsets of *G1B1toB6* that require training and testing on can be

calculated as Equation (6.3).

$$\begin{aligned}
 Num_{subset} &= \binom{N}{1} + \binom{N}{2} + \dots + \binom{N}{N} \\
 &= \sum_{1 \leq k \leq N} \binom{N}{k} \\
 &= \sum_{0 \leq k \leq N} \binom{N}{k} - \binom{N}{0} \\
 &= 2^N - 1
 \end{aligned} \tag{6.3}$$

Here, N denotes the number of features in $G1B1toB6$.

According to Equation (6.3), Table 6.13 calculated the numbers of subsets for exhaustive search (ES) on the feature combination $G1B1toB6$ under different cases. For different augmentation datasets with rectangle-based or circle-based Multi-Region Binarization, the number of features passing the boxplot check were different. The number of subsets, i.e., the number of the times of required training and testing, were more than 500 and close to 10000 sometimes for ES, indicating that the computational power needed for ES was much higher than the proposed algorithms (training and testing: 1 time required for EHHOG method, 5 times required for BiGHAM method, and 1 time required for iBiGHAM method) Thus, ES was plotted for reference only, but not considered to be used in the algorithms aiming at portable, real-time applications.

TABLE 6.13: Number of subsets for ES

ES: exhaustive search

Number of features in original feature set	Number of subsets
9	511
10	1023
11	2047
12	4095
13	8191

In Figures 6.8, 6.9, it can be observed that simply changing the rectangle-based region partition method to circle-base method or simply using the feature combination $G1B1ARSO$ instead of $G1B1B\alpha$ may not always improve the classification accuracy from that using the BiGHAM method ($G1B1B\alpha$ with rectangle-based Multi-Region Binarization), but using

G1B1ARSOCO and circle-based region partition method together increased the testing accuracy at most of the time. This is consistent with the classification results discussed in Section 6.4.1, proving again the usefulness of the proposed iBiGHAM method (*G1B1ARSOCO* with circle-based Multi-Region Binarization).

FIGURE 6.8: Testing accuracies of feature combination *G1B1B α* and *G1B1ARSOCO* with RBF SVM on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in *G1B1B2B3B4B5B6*, denoted by *G1B1toB6* for short. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 6.9: Testing accuracies of feature combination $G1B1B\alpha$ and $G1B1ARSO$ with BDT on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. BDT: bagging decision trees. The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$, denoted by $G1B1toB6$ for short. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

In Figure 6.10, the testing accuracies from using EHHOG method, to BiGHAM method, to iBiGHAM method were increasing and getting closer to the highest possible accuracy of exhaustive search (ES). Generally, the accuracies using bagging decision trees (BDT) were better than using radial basis function (RBF) support vector machine (SVM) as the classifier, but for small datasets (number of images per class ≤ 100) both classifiers had similar performances.

FIGURE 6.10: Testing accuracies of proposed EHHOG, BiGHAM, and iBiGHAM methods with RBF SVM and BDT on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. BDT: bagging decision trees. EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. EHHOG method: using EHHOG from original greyscale images and AR, SO from Otsu binarized images (Chapter 4), can be regarded as the feature combination $G1B1$ as defined in Section 5.3. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Existing Microorganism Image Classification Algorithms

In Chapter 4, the proposed EHHOG method was compared with 3 existing algorithms for microorganism images classification: Kumar's method [43], Bueno's method [49], and Pedraza's method [73]. Kumar's method and Bueno's method were handcrafted-feature-based

with artificial neural network (ANN) [43], bagging decision trees (BDT) [49] as the classifier. Pedraza's method was deep-learning-based so did not involve an additional classifier. The accuracy of Kumar's method was quite low compared to other existing methods and the proposed EHHOG method, and its computational complexity was higher than other handcrafted-feature-based methods (Bueno's method, EHHOG method), so Kumar's method was not used anymore. The classification results comparing the proposed algorithms (best case) and the existing algorithms Bueno's method [49], and Pedraza's method [73] are shown in Tables 6.14, 6.15 below.

The computational complexity to test 1 image for each of the proposed algorithms was about 40 MFLOPs (million floating-point operations), for Bueno's method was about 60 MFLOPs, and for Pedraza's method was over 700 MFLOPs (see Table 4.8 and Section 5.4.1). Bueno's method was not stable on our datasets. The testing accuracies for small datasets (number of images per class ≤ 100) were very low, for larger datasets were relatively high, but still lower than other methods (Pedraza's method, and proposed EHHOG, BiGHAM, iBiGHAM methods). The deep-learning-based method (Pedraza's method) has high classification accuracies, but the computational cost was much higher than handcrafted-feature-based methods. The proposed methods had good classification performances while keeping the computational cost low.

TABLE 6.14: Classification results comparing the existing algorithms and the proposed algorithms on our dataset

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively. * denotes the proposed method. EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. EHHOG method: using EHHOG from original greyscale images and AR, SO from Otsu binarized images (Chapter 4), can be regarded as the feature combination $G1B1$ as defined in Section 5.3. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). This table is a combination of Tables 4.8, 5.7, and 6.6. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Microorganism images clas- sification algorithm	Bueno's [49]	Pedraza's [73]	EHHOG *	BiGHAM *	iBiGHAM *
Testing accu- racy (best case)	93.0%	93.9%	92.2%	93.4%	93.9%

TABLE 6.15: Classification results comparing the existing algorithms and the proposed algorithms on our dataset

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. Num. refers to the number of images per class. * denotes the proposed method. EHHOG: entropy of histogram of HOG. HOG: histogram of oriented gradients. AR: aspect ratio. SO: solidity. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. EHHOG method: using EHHOG from original greyscale images and AR, SO from Otsu binarized images (Chapter 4), can be regarded as the feature combination $G1B1$ as defined in Section 5.3. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). The results of the proposed method in this table are extracted from Tables 6.8, 6.9, 6.10, 6.11 and 6.12. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Num.	Microorganism images classification algorithm				
	Bueno's [49]	Pedraza's [73]	EHHOG *	BiGHAM *	iBiGHAM *
30	56.7%	83.3%	82.2%	82.2%	85.6%
70	73.3%	92.4%	88.1%	90.5%	90.0%
100	33.3%	95.7%	91.7%	92.3%	95.3%
300	93.4%	96.2%	93.8%	95.8%	97.0%
700	93.5%	96.6%	97.0%	98.3%	98.6%
1000	93.9%	98.8%	97.7%	98.3%	99.2%

6.5 Summary of Proposed iBiGHAM Algorithm

Chapter 5 presented a proposed microorganism classification algorithm, Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM), for our waterborne pathogen images that increased the classification accuracy than the proposed EHHOG method in Chapter 4. In this chapter, with the aim of future portable, real-time water monitoring applications, the improved-BiGHAM (iBiGHAM) method was proposed and presented. For BiGHAM method, the classifier needed to be trained and tested for 5 times for the feature combinations $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). For iBiGHAM method, only 1 time of training and testing was needed with the feature combination $G1B1ARSOCO$. The mutual-information-based minimal-Redundancy-Maximal-Relevance (mRMR) criterion was applied to select the suitable features into the feature combination. Other than directly ranking all the features with

mRMR criterion and training and testing the classifier for about 10 times, which was the conventional way to apply mRMR feature selection (FS), a new way was proposed based on the prior knowledge of the features and binarization results of our images. In details, the proposed feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) extracted from the original greyscale images (feature set $G1$ as defined in Table 5.3) and the morphological features (aspect ratio (AR), solidity (SO), compactness (CO)) from Otsu binarized images and passing the boxplot check (feature set $B1$ as defined in Table 5.1) were put into the selected feature set directly. Then an AR, a SO, and a CO was selected by mRMR criterion from the feature sets from Multi-Region binarized images (feature set $B\alpha(\alpha = 2, 3, 4, 5, 6)$ as defined in Table 5.1). (Section 6.2)

In BiGHAM method, the proposed Multi-Region Binarization method used rectangle-based region partitioning, which was intuitive and simple for rectangular images. It could often extract out the outer scattering patterns missed by Otsu binarization but sometimes had unnatural horizontal or vertical lines. Since the microorganism cells and scattering patterns looked like circles in our images, the region partition method based on circles were tried and compared with the rectangle-based method. For the binarized images, the images binarized by the two region partition methods were similar most of the time, but sometimes the circle-based method gave a smoother and more complete binarized object. (Section 6.3)

The classification results showed that the proposed iBiGHAM algorithm using feature combination $G1B1ARSOCO$ and circle-based Multi-Region Binarization improved the testing accuracy from BiGHAM algorithm, and reduced the algorithm complexity that only required 1 time of classifier training and testing instead of 5 times for BiGHAM algorithm. Applying mRMR ranking directly on all the features that required 10 times of classifier training and testing costs more time and computational power in FS but did not have better accuracy than BiGHAM method, proving the usefulness of proposing a new way to select the features (feature combination $G1B1ARSOCO$ generated). Additionally, the proposed algorithms and some existing microorganism image classification algorithms were applied on the augmented datasets with 30, 70, 100, 300, 700, 1000 images. The accuracies of BiGHAM method were higher than those of EHHOG method, and using iBiGHAM method further improved the accuracy. The existing Bueno's method [49] performed badly on small datasets (number of images per class ≤ 100), and the accuracies for larger datasets were better but still lower than our algorithms. The accuracies of the deep-learning-based Pedraza's method [73] were

similar or slightly lower than the proposed iBiGHAM method, but the deep learning model required a much larger computational cost. (Section 6.4)

Considering the good classification performance on our optofluidic waterborne pathogen images and the low computational cost requirement, the proposed iBiGHAM algorithm shows the potential for future portable, real-time water monitoring applications. The proposed algorithms were also applied to an open-sourced dataset to verify the possibility to be extended to other images (to be presented in Chapter 7).

Chapter 7

Adaptation and Application of Proposed Algorithms on Open-Sourced Data

7.1 Introduction and Pre-Processing on Open-Sourced Dataset BCO-DMO 670706

As discussed in Section 3.2, the open-sourced image dataset BCO-DMO 670706 [23] was used to verify the proposed algorithms. It contains 2 types of coccolithophorid algae (*Pleurochrysis carterae*) virion images captured by a transmission electron microscope (TEM): the concentrated culture supernatant, and the band extracted from an Optiprep density gradient loaded with an aliquot of the previous concentrated sample. To apply the microorganism image classification algorithms, the following pre-processing procedures were implemented.

- 1) Excluding the non-single-cell images.
- 2) Cropping out the text notes at the bottom of the original images.
- 3) Rescaling the images so that the direct magnifications are the same ($180000\times$).
- 4) Cropping the images so that they have the same size, and the virion located around the centre of the image.

Some image examples of dataset BCO-DMO 670706 after pre-processing are shown in Figure ???. The image size after pre-processing is $1696*1696$ pixels. The numbers of type 1 and type 2 images are 14 and 6 respectively.

7.2 Adjustment on Data Augmentation and Binarization Procedures

Our images (examples shown in Figure 3.2) have brighter (closer to white) scattering patterns and darker (closer to black) background, so in the binarized images, the white colour represents the object of interest, and the black colour represents the background (examples shown in Figures 5.6, 6.2, 6.3). On the contrary, the images in the open-sourced dataset (examples shown in Figure ??) have brighter background than the object of interest. Thus, adjustments were applied for data augmentation and image binarization according to this image appearance.

7.2.1 Padding Methods for Data Augmentation

Similar to Section 6.4.2, the dataset was augmented by image rotation and random selection to become datasets with 30, 70, 100, 300, 700, 1000 images per class (type). For our images, the rotated images were padded with black (intensity = 0) because the background is very dark. For the images in the open-sourced dataset BCO-DMO 670706 [23], the background is brighter than the central part, so the following padding methods were used.

- 1) Grey padding: images padded with a constant grey colour (intensity = 200).
- 2) Symmetric padding: images padded with mirror reflections.

7.2.2 Binarization Settings

Since previously the white on the binarized image was the object and the black was the background, and the images in the open-sourced dataset BCO-DMO 670706 [23] have a brighter outer part (background) than the central part (virion cell, the object of interest), a swapping of black and white was applied for the images in the open-sourced dataset after binarization. Unlike the images in our dataset, the background of images in the open-sourced dataset was very heterogeneous, so 2 options of binarization settings were tried for the images in the open-sourced dataset as below.

- 1) Binarization option 1: the same as the binarization applied for our images presented in Chapters 4, 5, and 6

- 2) Binarization option 2: replacing Otsu's threshold [24] with $(0.85 \times \text{Otsu's threshold})$, then adding the morphological erosion after swapping the black and white (before keeping the largest object)

7.2.3 Image Illustration

The examples of images binarized with different binarization methods, padding methods, and binarization options are displayed in Figures 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, and 7.8 below. It can be observed that using Otsu binarization directly (i.e., Otsu binarization with binarization option 1 in Figures 7.1, 7.3) kept a lot of background noise, showing the necessity of making adaptations on the binarization procedures to get the object of interest extracted clearly.

Comparing these figures of binarized images for the open-sourced dataset and the binarized images for our dataset (Figures 5.6, 6.2, and 6.3), Multi-Region Binarization had different effects. For our optofluidic images, there were 2 parts of the scattering patterns, a brighter circle-like or ellipse-like inner part and a dimmer ring-like or irregular part, and the Multi-Region Binarization could capture both parts completely when the outer dimmer part was missed by other existing binarization methods. For the microscopic image in the open-sourced dataset, there were no scattering patterns but a round microorganism cell (object of interest) in the centre and grey noisy background. Multi-Region Binarization has the better ability to remove the noisy background, especially when using binarization option 1. The effects for classification would be examined by experiments later in Section 7.3.

FIGURE 7.1: Image example: Binarization procedures of Type 2, #1 image with Otsu binarization method, grey padding, binarization option 1

FIGURE 7.2: Image example: Binarization procedures of Type 2, #1 image with Otsu binarization method, grey padding, binarization option 2

FIGURE 7.3: Image example: Binarization procedures of Type 2, #1 image with Otsu binarization method, symmetric padding, binarization option 1

FIGURE 7.4: Image example: Binarization procedures of Type 2, #1 image with Otsu binarization method, symmetric padding, binarization option 2

FIGURE 7.5: Image example: Binarization procedures of Type 2, #1 image with circle-based 3-Region binarization method, grey padding, binarization option 1

FIGURE 7.6: Image example: Binarization procedures of Type 2, #1 image with circle-based 3-Region binarization method, grey padding, binarization option 2

FIGURE 7.7: Image example: Binarization procedures of Type 2, #1 image with circle-based 3-Region binarization method, symmetric padding, binarization option 1

FIGURE 7.8: Image example: Binarization procedures of Type 2, #1 image with circle-based 3-Region binarization method, symmetric padding, binarization option 2

7.2.4 Boxplot Check Results

The boxplot check results of features extracted from the binarized images using different padding methods and binarization options explained in Sections 7.2.1, 7.2.2 are displayed in the tables below.

TABLE 7.1: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, no augmentation

For the original dataset without data augmentation, numbers of type 1, 2 images are 14, 6 respectively. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result					
	Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO
Otsu's method [24]	0	0	0	1	1	1
Multi-Region Binarization, 2 regions	0	0	0	0	1	1
Multi-Region Binarization, 3 regions	0	1	1	0	0	1
Multi-Region Binarization, 4 regions	0	0	0	0	0	0
Multi-Region Binarization, 5 regions	0	0	0	0	0	0
Multi-Region Binarization, 6 regions	0	0	0	0	0	0

TABLE 7.2: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, no augmentation

For the original dataset without data augmentation, numbers of type 1, 2 images are 14, 6 respectively. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result					
	Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO
Otsu's method [24]	0	0	0	1	1	1
Multi-Region Binarization, 2 regions	0	0	0	0	1	1
Multi-Region Binarization, 3 regions	0	1	1	0	0	1
Multi-Region Binarization, 4 regions	0	0	0	0	0	0
Multi-Region Binarization, 5 regions	0	0	0	0	0	0
Multi-Region Binarization, 6 regions	0	0	0	0	0	0

TABLE 7.3: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 30 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	0	1	1	0	0	1	0	1	1
3-Region	0	1	1	0	0	0	0	1	1	0	0	1
4-Region	0	0	0	0	0	0	0	1	1	0	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 7.4: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 30 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	0	1	1	0	0	1	0	1	1
3-Region	0	1	1	0	0	0	1	1	1	0	0	0
4-Region	0	0	0	1	0	0	0	1	0	1	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 7.5: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 70 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	0	0	1	0	0	0	0	1	1
3-Region	0	1	1	0	0	0	0	1	1	0	0	1
4-Region	0	0	0	0	0	0	0	0	0	0	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 7.6: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 70 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	0	1	0	0	0	0	0	1	1
3-Region	0	1	1	0	0	0	0	1	1	0	0	0
4-Region	0	0	0	1	1	0	0	0	0	0	0	0
5-Region	0	1	1	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	1	0	0	0

TABLE 7.7: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 100 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	0	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	1
4-Region	0	0	0	0	0	0	0	0	0	0	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	1	0	0	0	0	0	0	0	0	0

TABLE 7.8: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 100 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	0	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	0
4-Region	0	0	0	1	0	0	0	0	0	0	0	0
5-Region	0	1	1	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	1	0	0	0

TABLE 7.9: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 300 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	0	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	1
4-Region	0	0	0	0	0	0	0	0	0	0	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	1	0	0	0	0	0	0	0	0	0

TABLE 7.10: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 300 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	0	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	0
4-Region	0	0	0	1	0	0	0	0	0	0	0	0
5-Region	0	1	1	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	1	0	0	0

TABLE 7.11: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 700 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	1	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	1
4-Region	0	0	0	0	0	0	0	0	0	0	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	1	0	0	0	0	0	0	0	0	0

TABLE 7.12: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 700 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	0	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	0
4-Region	0	0	0	1	0	0	0	0	0	0	0	0
5-Region	0	1	1	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	1	0	0	0

TABLE 7.13: Boxplot check results of features extracted from Otsu binarized and rectangle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 1000 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	1	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	1
4-Region	0	0	0	0	0	0	0	0	0	0	0	0
5-Region	0	0	0	0	0	0	0	0	0	0	0	0
6-Region	0	0	1	0	0	0	0	0	0	0	0	0

TABLE 7.14: Boxplot check results of features extracted from Otsu binarized and circle-based Multi-Region binarized images on open-sourced dataset BCO-DMO 670706, 1000 images per type

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. AR: aspect ratio. SO: solidity. CO: compactness. Binar. opt. refers to the binarization option. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check.

Binarization method	Boxplot check result											
	Grey padding						Symmetric padding					
	Binar. opt. 1			Binar. opt. 2			Binar. opt. 1			Binar. opt. 2		
	AR	SO	CO	AR	SO	CO	AR	SO	CO	AR	SO	CO
Otsu's [24]	0	0	0	1	1	1	0	0	0	1	1	1
2-Region	0	1	1	1	1	1	0	0	0	1	1	1
3-Region	0	1	1	0	0	0	0	1	1	1	0	0
4-Region	0	0	0	1	0	0	0	0	0	0	0	0
5-Region	0	1	1	0	0	0	0	0	0	0	0	0
6-Region	0	0	0	0	0	0	0	0	1	0	1	0

7.3 Classification Results and Discussions

In Section 7.2.4, it can be observed that when the binarization option was 1, none of the features aspect ratio (AR), solidity (SO), and compactness (CO) from Otsu binarized images could pass the boxplot check, i.e., there was no feature in the feature set $B1$. In this case, the proposed EHHOG method (Chapter 4) using feature combination $G1B1$ became using $G1$ only. Thus, in the experiments on the open-sourced dataset BCO-DMO 670706 [23], only the proposed BiGHAM (Chapter 5) and iBiGHAM (Chapter 6) methods were applied.

The original dataset of the open-sourced dataset was too small (only 14 and 6 images for type 1 and 2 respectively), causing high randomness in classification (the standard deviations of the 10 folds were about 20%). Thus, investigations on the augmented datasets were more necessary and reliable.

7.3.1 Proposed Algorithms

The feature set $G1$ from greyscale images and feature sets $B1$, $B2$, $B3$, $B4$, $B5$, $B6$ from binarized images were extracted according to the definitions in Tables 5.1, 5.3. The classification

results with bagging decision trees (BDT) of using features extracted from greyscale images and binarized images separately and together were plotted in Figures 7.9, 7.10, 7.11, 7.12, 7.13, and 7.14.

Figure 7.9 shows that when using a single feature, the proposed feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients), the accuracies using symmetric padding in data augmentation were much higher than those using grey padding.

FIGURE 7.9: Testing accuracies of feature set G1 with BDT on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. DT: decision trees. BDT: bagging decision trees. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Figure 7.10 displays the best accuracies among $B\gamma$ ($\gamma = 1, 2, 3, 4, 5, 6$), i.e., $B1, B2, B3, B4, B5, B6$, with different padding methods, binarization options, and region partition methods for Multi-Region Binarization. No conclusion could be drawn of which padding method, binarization option, and region partition method worked the best when using the feature extracted from binarized images individually.

FIGURE 7.10: Testing accuracies of feature set $B\gamma$ with BDT on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. DT: decision trees. BDT: bagging decision trees. The results displayed are the best among $B\gamma$ ($\gamma = 1, 2, 3, 4, 5, 6$), i.e., $B1$, $B2$, $B3$, $B4$, $B5$, $B6$. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Figures 7.11, 7.12, 7.13, and 7.14 compared the classification accuracies of using features extracted from greyscale images and binarized images separately and together. As explained in Section 6.4.2, the results of the exhaustive search (ES) on all the features were plotted for reference. Using features from greyscale images and binarized images together ($G1B1B\alpha$, $G1B1ARSOCO$, and ES) achieved accuracies close to or equal to 100%, much higher than using features from greyscale images ($G1$) or binarized images ($B\gamma$) separately, proved again the usefulness of combining features extracted from greyscale images and binarized images for classification in the proposed algorithms. The classification results with radial basis function (RBF) support vector machines (SVMs) as the classifier lead to the same conclusions as using bagging decision trees (BDT), so not plotted here.

FIGURE 7.11: Testing accuracies of feature combinations $G1$, $B\gamma$, $G1B1B\alpha$, $G1B1ARSOCO$, and ES with BDT, grey padding, binarization option 1 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. DT: decision trees. BDT: bagging decision trees. The results displayed as $B\gamma$ are the best among $B\gamma$ ($\gamma = 1, 2, 3, 4, 5, 6$), i.e., $B1, B2, B3, B4, B5, B6$. The results displayed as $G1B1B\alpha$ are the best among $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$, denoted by $G1B1toB6$ for short. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 7.12: Testing accuracies of feature combinations $G1$, $B\gamma$, $G1B1B\alpha$, $G1B1ARSOCO$, and ES with BDT, grey padding, binarization option 2 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. DT: decision trees. BDT: bagging decision trees. The results displayed as $B\gamma$ are the best among $B\gamma$ ($\gamma = 1, 2, 3, 4, 5, 6$), i.e., $B1$, $B2$, $B3$, $B4$, $B5$, $B6$. The results displayed as $G1B1B\alpha$ are the best among $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$, denoted by $G1B1toB6$ for short. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 7.13: Testing accuracies of feature combinations $G1$, $B\gamma$, $G1B1B\alpha$, $G1B1ARSOCO$, and ES with BDT, symmetric padding, binarization option 1 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. DT: decision trees. BDT: bagging decision trees. The results displayed as $B\gamma$ are the best among $B\gamma$ ($\gamma = 1, 2, 3, 4, 5, 6$), i.e., $B1$, $B2$, $B3$, $B4$, $B5$, $B6$. The results displayed as $G1B1B\alpha$ are the best among $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$, denoted by $G1B1toB6$ for short. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 7.14: Testing accuracies of feature combinations $G1$, $B\gamma$, $G1B1B\alpha$, $G1B1ARSOCO$, and ES with BDT, symmetric padding, binarization option 2 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. DT: decision trees. BDT: bagging decision trees. The results displayed as $B\gamma$ are the best among $B\gamma$ ($\gamma = 1, 2, 3, 4, 5, 6$), i.e., $B1, B2, B3, B4, B5, B6$. The results displayed as $G1B1B\alpha$ are the best among $G1B1B\alpha$ ($\alpha = 2, 3, 4, 5, 6$). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$, denoted by $G1B1toB6$ for short. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

For a clearer viewing, the classification accuracies comparing the proposed Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) and improved-BiGHAM (iBiGHAM) methods were plotted in Figures 7.15, 7.16, 7.17, and 7.18. All these experiments achieved high classification accuracies (over 90%), showing the good classification ability of the proposed algorithms on the open-sourced dataset BCO-DMO 670706 [23]. From these results, using iBiGHAM method with the symmetric padding and binarization option 2 had the best performance with classification accuracies close to or equal to 100%.

In Figure 7.15, the accuracies of iBiGHAM method were generally higher than using BiGHAM method, similar to the classification results on our dataset in Section 6.4.2. In Figures 7.16, 7.17, and 7.18, the accuracies of BiGHAM and iBiGHAM methods were similar, all close to the reference of exhaustive search (ES) results.

FIGURE 7.15: Testing accuracies of proposed BiGHAM and iBiGHAM methods with RBF SVM and BDT, grey padding, binarization option 1 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. BDT: bagging decision trees. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 7.16: Testing accuracies of proposed BiGHAM and iBiGHAM methods with RBF SVM and BDT, grey padding, binarization option 2 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. BDT: bagging decision trees. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 7.17: Testing accuracies of proposed BiGHAM and iBiGHAM methods with RBF SVM and BDT, symmetric padding, binarization option 1 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. BDT: bagging decision trees. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

FIGURE 7.18: Testing accuracies of proposed BiGHAM and iBiGHAM methods with RBF SVM and BDT, symmetric padding, binarization option 2 on the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. RBF: radial basis function. SVM: support vector machine. BDT: bagging decision trees. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization (Chapter 6). The green curves are the reference of the best possible accuracies obtained by exhaustive search (ES) on all the features in $G1B1B2B3B4B5B6$. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

7.3.2 Existing Microorganism Image Classification Algorithms

Classification experiments using some existing microorganism image classification algorithms on the open-sourced dataset BCO-DMO 670706 [23] were done for comparison. The best-case testing accuracies of the existing algorithms and the proposed algorithms on our dataset and the open-sourced dataset are summarized in Table 7.15 and Table 7.16. As discussed in Section 7.3.1, for the open-sourced dataset, the proposed EHHOG algorithm was often not

applicable, and the accuracies of the proposed BiGHAM and iBiGHAM algorithms were similar under the best case (symmetric padding, binarization option 2 with bagging decision trees (BDT) as the classifier), so only the results of iBiGHAM algorithm with symmetric padding, binarization option 2 and BDT are presented here. The best case of applying the existing Bueno’s algorithm [49] on the open-sourced dataset was using symmetric padding with BDT. The best case of applying the existing Pedraza’s algorithm [73] on the open-sourced dataset was using symmetric padding for the 30, 70 image-per-class cases, and grey padding for other cases. The proposed algorithms and Bueno’s algorithm are handcrafted-feature-based methods. Pedraza’s algorithm is a deep-learning-based method.

For the original dataset, our dataset has 147, 117, 148 images for the 3 classes, and the open-sourced dataset BCO-DMO 670706 only has 14, 6 images for the 2 classes (types). The classification accuracies on the original open-sourced dataset were relatively low with high standard deviations, indicating that a sufficient number of samples (images) were important for classification. Thus, classification experiments with the augmented datasets were more necessary and reliable.

TABLE 7.15: Classification results comparing the existing algorithms and the proposed algorithm on our dataset and the open-sourced dataset BCO-DMO 670706

For the original dataset without data augmentation, numbers of class G, C, O images are 147, 117, 148 respectively for our dataset. For the original dataset without data augmentation, numbers of type 1, 2 images are 14, and 6 respectively for the open-sourced dataset BCO-DMO 670706. * denotes the proposed method. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. iBiGHAM method: using feature combination *G1B1ARSOCO* with circle-based Multi-Region Binarization (Chapter 6). The results on our dataset are from Table 6.14. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Dataset	Microorganism images classification algorithm		
	Bueno’s [49]	Pedraza’s [73]	iBiGHAM *
Our dataset	93.0%	93.9%	93.9%
Open-sourced dataset BCO-DMO 670706 [23]	70.0%	50.0%	85.0%

For the augmented datasets, more image samples usually gave higher classification accuracies. Generally, the proposed iBiGHAM algorithm had good classification accuracies under

different conditions. For our dataset, Bueno's algorithm [49] did not work well, especially for small datasets (number of images per class ≤ 100), and the accuracies using Pedraza's algorithm [73] and the proposed iBiGHAM algorithm were similar. But for the open-sourced dataset, the performance of Pedraza's algorithm was not as good as Bueno's algorithm or the proposed iBiGHAM algorithm. The reason could be that the images in the open-sourced dataset have noisy backgrounds, so the handcrafted-feature-based methods had advantages over the deep-learning-based methods when doing the feature extraction focusing on the object of interest.

TABLE 7.16: Classification results comparing the existing algorithms and the proposed algorithm on our dataset and the open-sourced dataset BCO-DMO 670706

For the augmented datasets, original images are rotated to generate more than 1000 images and then randomly selected certain numbers. Num. refers to the number of images per class. * denotes the proposed method. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. iBiGHAM method: using feature combination *G1B1ARSOCO* with circle-based Multi-Region Binarization (Chapter 6). The results on our dataset are from Table 6.15. With 10-fold cross-validation, the testing accuracies displayed are the mean value of the 10 folds.

Dataset	Num.	Microorganism images classification algorithm		
		Bueno's [49]	Pedraza's [73]	iBiGHAM *
Our dataset	30	56.7%	83.3%	85.6%
	70	73.3%	92.4%	90.0%
	100	33.3%	95.7%	95.3%
	300	93.4%	96.2%	97.0%
	700	93.5%	96.6%	98.6%
	1000	93.9%	98.8%	99.2%
Open-sourced dataset BCO-DMO 670706 [23]	30	96.7%	80.0%	98.3%
	70	95.7%	81.4%	100%
	100	100%	82.5%	100%
	300	99.8%	86.0%	100%
	700	100%	89.3%	100%
	1000	100%	92.4%	100%

As mentioned in Section 6.4.2, the computational complexity to test 1 image for the proposed algorithm was about 40 MFLOPs (million floating-point operations), for Bueno's algorithm was about 60 MFLOPs, and for Pedraza's algorithm was more than 700 MFLOPs. The number of features for the proposed algorithm was no more than 7, for Bueno's algorithm was about 100 or 200, for Pedraza's algorithm was 4096. Considering the computational cost and the classification performance on different datasets, the proposed iBiGHAM algorithm shows good microorganism classification ability that has the potential for future portable, real-time applications.

7.4 Summary

To verify the classification performances on other images, the proposed and existing algorithms were implemented on the virion images of an open-sourced dataset BCO-DMO 670706 [23]. Pre-processings were done to remove the text notes and unify the image magnification and size. The selected and pre-processed single-cell images were used for classification. (Section 7.1)

The images in different datasets usually have different characteristics, so adjustments were made on padding methods for data augmentation and binarization procedures. For the images in our dataset, the most inner part is the microorganism cell close to a black round, then the scattering patterns including a brighter ellipse-like inner part and a dimmer irregular part, the background is all black or close to black. For the images in the open-sourced dataset, the microorganism cell is darker, the background is bright (grey) and noisy. Thus, swapping of the black and white was applied after the image binarization for the open-sourced dataset. For data augmentation, instead of padding with black (intensity = 0) in image rotation for our images, 2 padding methods were tried for the images in the open-sourced dataset: grey padding (padding with intensity = 200) and symmetric padding (padding with mirror reflections). To binarize out the object of interest clearly, besides the same binarization procedures used for our images (called binarization option 1), another binarization settings (called binarization option 2) using 0.85 times of Otsu's threshold and adding a morphological erosion after swapping the black and white. The Otsu binarized images using binarization option 1 kept much background noise, and the corresponding features could not pass the boxplot check, proving the necessity of binarization adaptations. (Section 7.2)

In the classification experiments on the open-sourced dataset, the accuracies of combining features extracted from greyscale images and binarized images were much higher than using them separately, similar to the conclusion from the experiments on our dataset, verifying the proposed algorithms. Sometimes, the features extracted from Otsu binarized images did not pass the boxplot check, and the proposed EHHOG algorithm combining the feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) from greyscale images and morphological features from Otsu binarized images was not applicable. Nevertheless, the proposed BiGHAM (Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization) and iBiGHAM (improved-BiGHAM) algorithms worked well on the open-sourced dataset. Especially, with symmetric padding and binarization option 2, the accuracies of BiGHAM and iBiGHAM algorithms with 30, 70, 100, 300, 700, 1000 images per class were all close or equal to 100%. As for the existing microorganism classification algorithms, the handcrafted-feature-based Bueno's algorithm [49] performed fine on the open-sourced dataset but not on our dataset, the deep-learning-based Pedraza's algorithm [73] performed fine on our dataset but not on the open-sourced dataset. The handcrafted-feature-based proposed iBiGHAM algorithm outperformed them for both datasets in most cases. Besides, the proposed method has a much smaller computational requirement and feature number, so very suitable for portable, real-time microorganism classification applications not only with optofluidic images but also microscopic images. (Section 7.3)

Chapter 8

Conclusions and Future Outlook

8.1 Conclusions

Pathogenic microorganisms in the water form an unignorable risk to human well being, so microorganism detection is essential for water quality control. To develop computer-based methods that are simpler, faster, cheaper, and more environmentally friendly than the conventional biochemical based microorganism detection methods, this PhD study focused on classifying the waterborne pathogens using our dataset of optofluidic images that had the potential for portable, real-time water monitoring applications. The dataset contains single-cell images of *Giardia*, *Cryptosporidium* (2 common waterborne pathogens), and other non-hazardous microorganism particles. Most existing microorganism classification algorithms were designed for images taken from microscopes, which were much larger and expensive than an optofluidic imaging system. The optofluidic images had different appearances from the microscopic images, so new classification algorithms were needed. With the aim of future portable, real-time applications, the algorithms were designed under the considerations of improving the classification performance and keeping the computational cost low together.

Chapter 4 presented the proposed EHHOG method using 3 features: the proposed feature EHHOG (entropy of histogram of HOG) (HOG: histogram of oriented gradients) from the original greyscale images, some morphological feature (aspect ratio (AR) and solidity (SO)) from the Otsu binarized images for classification. Through the mathematical derivations, figure plots, and classification experiments, the proposed feature EHHOG was proved to contribute much to classify our images for it summarized the local edge appearances of the whole image which could differentiate out the class G (*Giardia*) images from other 2 classes. The radial basis function (RBF) support vector machine (SVM) and bagging decision trees

(BDT) were 2 classifiers that gave high accuracies in the experiments, so used throughout the study. Using the proposed EHHOG method with BDT, the accuracy achieved 92.2% on our dataset, much higher than using the existing Kumar's method [43]. The existing Bueno's method [49] with linear SVM and Pedraza's method [73] had the accuracies of 93.0% and 93.9% respectively but required higher computational cost. Especially, the deep-learning-based Pedraza's method had more than 15 times of computational complexity (number of FLOPs (floating-point operations) to test one image) than EHHOG method. The proposed EHHOG method gave a realistic solution to classify microorganism images when taking care of both the classification performance and the computational cost.

Chapter 5 presented the proposed Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) method using the feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization to improve the classification. In our optofluidic images, the scattering patterns could be regarded as 2 parts: a brighter inner part and a dimmer outer part. When using the conventional global thresholding Otsu's binarization method [24], the outer part was usually missed. To capture the missed information, a local thresholding Multi-Region binarization method was proposed. Different feature combinations were investigated and $G1B1B\alpha$ combining the features from the original greyscale images ($G1$), the features from Otsu binarized images ($B1$), and the features from Multi-Region Binarized images ($B\alpha$ $\alpha = 2, 3, 4, 5, 6$) was finally chosen with higher accuracies than using EHHOG method and similarly low computational complexity to test one image. The accuracy of the proposed BiGHAM using the feature combination $G1B1B\alpha$ with an RBF SVM achieved 93.4%, higher than using the existing Bueno's method. Also, if using some existing binarization methods (Sauvola's method [101], K-means based method [102]) instead of the proposed Multi-Region Binarization, the accuracies dropped below 90%, again proved the usefulness of the proposed Multi-Region Binarization for classification.

Chapter 6 presented the proposed improved-BiGHAM (iBiGHAM) method using the feature combination $G1B1ARSOCO$ with circle-based Multi-Region Binarization to improve the classification. The feature combination $G1B1ARSOCO$ was a combination of the features from the original greyscale images ($G1$), the features from Otsu binarized images ($B1$), and 1 aspect ratio (AR), 1 solidity (SO), 1 compactness (CO) selected from Multi-Region Binarized images using a mutual-information-based feature selection (FS). Besides the rectangle-based region partition method which was intuitive and simple for rectangular images, circle-based

regions were also implemented for Multi-Region Binarization. Using the feature combination *G1B1ARSO*CO with circle-based Multi-Region Binarization improved the classification accuracy to 93.9% from 93.4% of BiGHAM, the same as the deep-learning-based Pedraza's method with a much lower computational requirement. Furthermore, the feature combination *G1B1ARSO*CO did not require training and testing for 5 times as the feature combination *G1B1B* α ($\alpha = 2, 3, 4, 5, 6$), so simplified the algorithm. Thus, the method was called improved-BiGHAM (iBiGHAM) method. Other than the original dataset with 147, 117, 148 images for class G, C, O respectively, experiments were also done on the augmented datasets (images rotated and randomly selected) with 30, 70, 100, 300, 700, 1000 images per class. The existing Bueno's method did not get good accuracies, specially for small datasets (number of images per class ≤ 100). The accuracies were generally increasing from the proposed EHHOG method to the proposed BiGHAM method to the proposed iBiGHAM method. Using iBiGHAM method, the accuracies could achieve the same level as using the existing Pedraza's method, while Pedraza's method is deep-learning-based and thus required much higher computing resources and time.

To verify the proposed and existing classification algorithms, the open-sourced dataset BCO-DMO 670706 [23] of virion images taken by a microscope was used. Based on the appearance difference between our images and the images in the open-sourced dataset, some adaptations were made for the data augmentation and image binarization procedures. The proposed EHHOG method was not applicable sometimes, but both BiGHAM and iBiGHAM methods kept achieving good classification accuracies. Different from the experimental results on our dataset, the accuracies using the existing Pedraza's method were relatively low, and the accuracies using the existing Bueno's method and the proposed BiGHAM/iBiGHAM method were at the same level. In summary, Bueno's method did not work well for our dataset, Pedraza's method did not work well for the open-sourced dataset BCO-DMO 670706, while the proposed BiGHAM and iBiGHAM methods worked well for both datasets. Considering the low computational cost, the proposed algorithms showed good potential for future portable, real-time water monitoring applications and even other applications on various types of microorganism images.

8.2 Recommendations for Future Work

8.2.1 Deep Learning Approaches

Deep learning is a kind of machine learning approach that uses multiple layers of nonlinear processing (e.g., a multi-layer artificial neural network (ANN)) [36],[65]–[67]. When used for image classification, it does not require handcrafting features for specific image datasets. For a deep learning model, the input is the image and the output is the predicted class label, so it can be thought of as the combination of a feature extractor and a classifier. To train a deep learning model from the beginning, large datasets (e.g., more than 1000 images per class) and high computational power are necessary [68].

In microorganism classification, the number of images is often limited by the cost (long time, expensive instruments, professional personals, etc.) of capturing microorganism images, so many applications of deep learning on microorganism image classification were based on fine-tuning a deep learning model that has been pre-trained on other large image datasets [19]. For example, the existing method [73] that we compared with was using a pre-trained deep learning convolutional neural network (CNN) model (pre-trained AlexNet [72]) to classify microscopic diatom images.

During fine-tuning, different parameters/settings of the deep learning model can be changed, e.g., the learning rate, the optimization method, the batch size, the number of epochs, the dropout values, etc. . Then the model with different settings will be trained and tested on the microorganism dataset to see which setting gives highest testing accuracy. The best setting of this model for the specific dataset is given by trials in this way. When we applied this method [73] on our dataset and the open-sourced dataset BCO-DMO 670706 [23], a preliminary investigation was made on the settings of the deep learning model: the combinations of different dropout values in the fully-connected (FC) layer (the core of the CNN's classifier part) and different learning rates for the CNN model were tried. As the limitation of time and computing resources, the following settings were fixed: using stochastic gradient descent (SGD) for optimization during training and using 100 as the batch size (details explained in Appendix E).

- Dropout of the FC layer
 - 30%
 - 50%
 - 70%

- Learning rate (LR)
 - Fixed LR, 0.05
 - Fixed LR, 0.01
 - Fixed LR, 0.005
 - Fixed LR, 0.001
 - Step LR, initial at 0.05, decrease to 0.8 of the previous LR with every 10 epochs
 - Step LR, initial at 0.05, decrease to 0.8 of the previous LR with every 20 epochs
 - Step LR, initial at 0.01, decrease to 0.8 of the previous LR with every 10 epochs
 - Step LR, initial at 0.01, decrease to 0.8 of the previous LR with every 20 epochs
 - Step LR, initial at 0.005, decrease to 0.8 of the previous LR with every 10 epochs
 - Step LR, initial at 0.005, decrease to 0.8 of the previous LR with every 20 epochs
 - Step LR, initial at 0.001, decrease to 0.8 of the previous LR with every 10 epochs
 - Step LR, initial at 0.001, decrease to 0.8 of the previous LR with every 20 epochs

Figure 8.1 shows how the step learning rate (LR) changes with epochs.

FIGURE 8.1: Illustration of step learning rate (LR) initial at 0.05, decrease to 0.8 of the previous LR with every 20 epochs

Fine-Tuning Settings for Deep Learning on Our Dataset

Figure 8.2, 8.3, and 8.4 display the classification results. For our dataset, there are 3 classes of images and a random guess will give $1/3$ accuracy, so the plots start from 0.3. It can be observed that the accuracies with fixed learning rate (LR) 0.05 and step LR's initialized at 0.05 were the same as the random guess. The accuracies with fixed LR 0.01 and step LR's initialized at 0.01 were better but generally low. For the other LR's, the accuracies were generally over 80%, but there is no rule for which one performed better. For example, with 30% dropout, the accuracies using fixed LR 0.005 were obviously lower than those using fixed LR 0.001, but with 50% or 70% dropout, the 2 LR's performed similarly.

FIGURE 8.2: Testing accuracies of using the DL-based method [73] to classify our images with different learning rates, dropout = 30%

DL-based: deep-learning-based.

FIGURE 8.3: Testing accuracies of using the DL-based method [73] to classify our images with different learning rates, dropout = 50%

DL-based: deep-learning-based.

FIGURE 8.4: Testing accuracies of using the DL-based method [73] to classify our images with different learning rates, dropout = 70%

DL-based: deep-learning-based.

Among the curves in the 3 figures above, the setting "dropout = 50%, step learning rate (LR) initialized at 0.001, decreased to 0.8 of the previous LR with every 10 epochs" was used when comparing our method and the deep learning method [73] on our dataset for this setting performed well at all conditions of numbers of samples (images) per class.

Fine-Tuning Settings for Deep Learning on Open-sourced Dataset BCO-DMO 670706

There were 2 padding methods used based on the image appearances in the open-sourced dataset BCO-DMO 670706 [23]: padding with a grey (intensity = 200) colour (called grey padding), and padding with mirror reflections (called symmetric padding) when augmenting the dataset with image rotation.

Figure 8.5, 8.6, 8.7, 8.8, 8.9, and 8.10 display the classification results. For the open-sourced dataset, there are 2 classes of images and a random guess will give 1/2 accuracy, so the plots start from 0.5. It can be observed that the accuracies with fixed learning rate (LR) 0.05 and step LRs initialized at 0.05 were the same as the random guess (except for the step LRs initialized at 0.05 with 50% dropout, they performed much better than random guessing as other LRs). For the other LRs, the accuracies were generally higher, but there is no rule for which one performed better. For example, the accuracies using fixed LR 0.005 with 30% or 70% dropout were relatively high, but those with 50% dropout were quite low, and even dropped to 50% (same as random guessing) for the 1000-image-per-class case.

It can also be observed that, for small datasets with less than 100 images per class, the accuracies with symmetric padding were higher, but for other cases, the accuracies with grey padding were higher. This was quite different from the results applying handcrafted-feature-based methods on the open-sourced dataset, where symmetric padding always performed better.

FIGURE 8.5: Testing accuracies of using the DL-based method [73] to classify the open-sourced dataset images with different learning rates, dropout = 30%, grey padding

DL-based: deep-learning-based.

FIGURE 8.6: Testing accuracies of using the DL-based method [73] to classify the open-sourced dataset images with different learning rates, dropout = 50%, grey padding

DL-based: deep-learning-based.

FIGURE 8.7: Testing accuracies of using the DL-based method [73] to classify the open-sourced dataset images with different learning rates, dropout = 70%, grey padding

DL-based: deep-learning-based.

FIGURE 8.8: Testing accuracies of using the DL-based method [73] to classify the open-sourced dataset images with different learning rates, dropout = 30%, symmetric padding

DL-based: deep-learning-based.

FIGURE 8.9: Testing accuracies of using the DL-based method [73] to classify the open-sourced dataset images with different learning rates, dropout = 50%, symmetric padding

DL-based: deep-learning-based.

FIGURE 8.10: Testing accuracies of using the DL-based method [73] to classify the open-sourced dataset images with different learning rates, dropout = 70%, symmetric padding

DL-based: deep-learning-based.

Among the curves in the 6 figures above, the setting "dropout = 30%, fixed learning rate (LR) initialized at 0.005, symmetric padding for small datasets (less than 100 images per class), grey padding for other cases" was used when comparing our method and the deep

learning method [73] on the open-sourced dataset BCO-DMO 670706 [23] for this setting performed well at all conditions of numbers of samples (images) per class.

Lightweight Deep Learning Models

During recent years, some lightweight deep learning (DL) models with relatively low computational complexities among DL models were also developed for mobile applications. They are also a possible direction for the future study. The following discussions show two examples of lightweight DL models that can be considered for microorganism classification.

- **MobileNet.** MobileNet model (MobileNetV1 [103]: 569 MFLOPs; MobileNetV2 [104]: 300 MFLOPs; MobileNetV3 [105]: 219 MFLOPs) was designed based on the trade-off between accuracy and model size/ latency. It has smaller computational complexity than other popular DL models (AlexNet [72]: 716 MFLOPs; VGG-16 [74]: 15300 MFLOPs; GoogLeNet [75]: 1550 MFLOPs). Some preliminary results of applying pre-trained MobileNet on our dataset were shown in Appendix E.3 but there was no conclusion about the optimal fine-tuning setting and the best performance that MobileNet can achieve. In the future, if the time and computing resources allow, more studies may be done about MobileNet applications on microorganism classification.
- **U-Net.** U-Net [106],[107] (62 MFLOPs) was a small DL model designed and pre-trained for complicated semantic segmentation tasks, especially popular in the field of biomedical image segmentation for its efficiency on the utilization of annotated data. For the U-Net applications on image classification, the U-Net was usually used for sophisticated multiple-object image segmentation, followed by rule derivation or feature extraction and classifiers [108],[109]. Currently, the proposed classification algorithms use the binarization (a special case of segmentation where there is only one single object and the background to be separated) methods based the simple and fast Otsu's thresholding and the binarization results are good enough for further feature extraction and classification, which also shows an advantage of optofluidic imaging over microscopic imaging (easy to obtain single-cell images for classification, discussed in Section 2.1). Besides, the images used in this PhD study are single-cell and with scattering patterns, and there is no clinical or microbiological meaning on the segmentation ground truth as in the U-Net applications [108],[109]. Thus, the proposed algorithms were evaluated

on the classification performance, and only used simple thresholding-based image binarization. In the future, if more types of microorganisms or fluid samples are included and the images require sophisticated segmentation, U-Net will be a good direction for study.

Summary

The complete experiment results were displayed in Appendix E. For the aim of potential portable real-time water monitoring applications, the deep learning methods are not very suitable for the high computational cost, and sometimes their accuracies were even much lower than those using handcrafted-feature-based methods (Chapter 7).

However, for general microorganism image classification, how to determine the settings of deep learning models is an interesting and useful topic. Currently, most deep learning applications designed the learning rate, the optimization method, the batch size, the number of epochs, the dropout values, etc. manually by trials [72],[73],[110]. As shown in the experiment results above, different settings resulted in very different accuracies, the model with some settings even failed to classify (accuracies same as random guessing). Some learning rates (LRs) worked well with some dropout value, but the accuracies decreased a lot with another dropout value. And for different datasets, the optimal LR and dropout combination was also quite different. For the open-sourced dataset, the performance trends of the 2 padding methods were quite different from the handcrafted-feature-based methods. In the future, a possible research direction would be the reason for these results and how to automatically determine a good setting for the deep learning models.

Besides, the lightweight deep learning (DL) models may also be investigated for microorganism detection applications.

8.2.2 Water Quality Monitoring and Other Applications

In this PhD study, classification algorithms proposed based on optofluidic waterborne pathogen images aiming at portable, real-time applications. To develop such water quality monitoring applications, future work can be considered in the following directions.

- **Investigating more types of waterborne microorganisms.** Due to the limit of time, personnel, materials and instruments, our dataset only included 2 common waterborne

pathogenic protozoa, *Giardia*, *Cryptosporidium*, and 1 class of non-hazardous particles in the water. For future water quality monitoring applications, more types of water-borne pathogens and particles could be involved and classification algorithms could be developed accordingly.

- **Optimizing the current algorithm in a more automatic way.** The proposed algorithms had some settings determined manually from the prior knowledge or experimental trials, such as the way to apply the mutual-information-based feature selection (FS) in the proposed iBiGHAM (improved-BiGHAM) (BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization) algorithm was based on the prior knowledge of the appearances of greyscale and binarized images. And the region partition method, parameters in the threshold calculation, number of regions used were based on the trials of binarization to capture the object of interest clearly. In the future, when more types of microorganisms were involved in the study, a more automatic way to determine the settings may be in need. Other features, FS methods, and classifiers may also be studied.
- **Unsupervised learning.** The current proposed algorithms were based on supervised learning benefited from the available ground truth labels. In the future, more challenging tasks like semi-supervised learning (partial labels) and unsupervised learning (no labels) may be explored so that the computational cost for training can be reduced.
- **Algorithm robustness.** The robustness of the algorithms in real applications can also be discussed. For examples, if more images can be captured and included in the future, the classification performances for non-targeted microorganism types, non-single-cell images, and low-resolution/noisy images can be examined.
- **Embedding the algorithm onto hardware.** Smartphones are feasible and popular devices for portable machine learning applications as they are readily accessible all over the world [111]. Many image classification applications are deployed on smartphones with cloud computing [112],[113], but necessary Internet connection for those applications may be unavailable in rural or undeveloped areas, where the people suffered more by waterborne pathogen infections [2]. Thus, for portable and real-time water-borne pathogen detection, more feasible approaches of hardware implementations are

an optofluidic attachment on a smartphone [111] or a smartphone (mobile app for image classification) and an optofluidic device wireless communicated via Bluetooth [114].

The experiments on the open-sourced dataset BCO-DMO 670706 [23] show that the proposed algorithms can also be applied to other images (e.g., the microscopic virion images) with some simple adaptations. Thus, extensions to applications in other microorganism investigation fields, like food and beverage analysis, wastewater treatment, and blood testing, are also possible and promising.

Author's Publications

- 1) J. Luo, and W. Ser, "Image Classification of Microorganisms Using Mutual Information Approach," in *IEEE International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018. (Abstract track)
- 2) J. Luo, A. Liu, P.H. Yap, B. Liedberg, and W. Ser, "An mRMR-SVM Approach for Opto-Fluidic Microorganism Classification," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 666-669, IEEE, 2018.
- 3) J. Luo, W. Ser, A. Liu, P.H. Yap, B. Liedberg, and S. Rayatpisheh, "Waterborne Pathogen Classification with Optofluidic Images," *Biomedical Signal Processing and Control*. (Submitted)
- 4) J. Luo, W. Ser, A. Liu, P.H. Yap, B. Liedberg, and S. Rayatpisheh, "Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization (BiGHAM) for Microorganism Image Classification," *Multidimensional Systems and Signal Processing*. (Submitted)
- 5) J. Luo, W. Ser, A. Liu, P.H. Yap, B. Liedberg, and S. Rayatpisheh, "Microorganism Image Classification with Circle-Based Multi-Region Binarization and Mutual-Information-Based Feature Selection," *Biomedical Engineering Advances*, vol. 2, pp. 100020, 2021.

Bibliography

- [1] WHO, “Guidelines for drinking-water quality,” *WHO chronicle*, vol. 38, no. 4, 2011. World Health Organization.
- [2] L. Savioli, H. Smith, and A. Thompson, “Giardia and cryptosporidium join the ‘neglected diseases initiative’,” *Trends in parasitology*, vol. 22, no. 5, pp. 203–208, 2006.
- [3] C. L. Fischer Walker, M. J. Aryee, C. Boschi-Pinto, and R. E. Black, “Estimating diarrhea mortality among young children in low and middle income countries,” *PloS one*, vol. 7, no. 1, p. e29151, 2012.
- [4] M. I. Z. Sato, A. T. Galvani, J. A. Padula, A. C. Nardocci, M. de Souza Lauretto, M. T. P. Razzolini, and E. M. Hachich, “Assessing the infection risk of giardia and cryptosporidium in public drinking water delivered by surface water systems in sao paulo state, brazil,” *Science of The Total Environment*, vol. 442, pp. 389–396, 2013.
- [5] S. A. Squire and U. Ryan, “Cryptosporidium and giardia in africa: current and future challenges,” *Parasites & vectors*, vol. 10, no. 1, pp. 1–32, 2017.
- [6] G. Certad, E. Viscogliosi, M. Chabé, and S. M. Cacciò, “Pathogenic mechanisms of cryptosporidium and giardia,” *Trends in Parasitology*, vol. 33, no. 7, pp. 561–576, 2017.
- [7] Y. A. L. Lim, R. Ahmad, and H. Smith, “Current status and future trends in cryptosporidium and giardia epidemiology in malaysia,” *Journal of water and health*, vol. 6, no. 2, pp. 239–254, 2008.
- [8] E. A. Dreelin, R. L. Ives, S. Molloy, and J. B. Rose, “Cryptosporidium and giardia in surface water: a case study from michigan, usa to inform management of rural water systems,” *International journal of environmental research and public health*, vol. 11, no. 10, pp. 10480–10503, 2014.

- [9] F. Y. Ramírez-Castillo, A. Loera-Muro, M. Jacques, P. Garneau, F. J. Avelar-González, J. Harel, and A. L. Guerrero-Barrera, “Waterborne pathogens: detection methods and challenges,” *Pathogens*, vol. 4, no. 2, pp. 307–334, 2015.
- [10] P. Ligda, E. Claerebout, D. Kostopoulou, A. Zdragas, S. Casaert, L. J. Robertson, and S. Sotiraki, “Cryptosporidium and giardia in surface water and drinking water: Animal sources and towards the use of a machine-learning approach as a tool for predicting contamination,” *Environmental Pollution*, vol. 264, p. 114766, 2020.
- [11] M. R. Salton and K.-S. Kim, *Medical microbiology*. University of Texas Medical Branch at Galveston, 4th edition ed., 1996. Chapter 2 Structure (under Bacteriology).
- [12] H. R. Gelderblom, *Medical microbiology*. University of Texas Medical Branch at Galveston, 4th edition ed., 1996. Chapter 41 Structure and Classification of Viruses (under Virology).
- [13] R. G. Yaeger, *Medical microbiology*. University of Texas Medical Branch at Galveston, 4th edition ed., 1996. Chapter 77 Protozoa: Structure, Classification, Growth, and Development (under Parasitology).
- [14] D. Wakelin, *Medical microbiology*. University of Texas Medical Branch at Galveston, 4th edition ed., 1996. Chapter 87 Helminths: Pathogenesis and Defenses (under Parasitology).
- [15] L. F. Strenkoski, M. A. Schneider, S. C. Swamy, and M. S. Nagar, “Rapid microorganism detection method,” Dec. 1 1998. US Patent 5,843,699.
- [16] O. Lazcka, F. J. Del Campo, and F. X. Munoz, “Pathogen detection: A perspective of traditional methods and biosensors,” *Biosensors and bioelectronics*, vol. 22, no. 7, pp. 1205–1217, 2007.
- [17] P. Mandal, A. Biswas, K. Choi, and U. Pal, “Methods for rapid detection of foodborne pathogens: an overview,” *American Journal Of Food Technology*, vol. 6, no. 2, pp. 87–102, 2011.
- [18] N. A. Mohamad, N. A. Jusoh, Z. Z. Htike, S. L. Win, *et al.*, “Bacteria identification from microscopic morphology: a survey,” *International Journal on Soft Computing, Artificial Intelligence and Applications (IJSCAI)*, vol. 3, no. 1, pp. 2319–1015, 2014.

- [19] C. Li, K. Wang, and N. Xu, “A survey for the applications of content-based microscopic image analysis in microorganism classification domains,” *Artificial Intelligence Review*, vol. 51, no. 4, pp. 577–646, 2019.
- [20] D. Psaltis, S. R. Quake, and C. Yang, “Developing optofluidic technology through the fusion of microfluidics and optics,” *nature*, vol. 442, no. 7101, pp. 381–386, 2006.
- [21] X. Fan and I. M. White, “Optofluidic microsystems for chemical and biological analysis,” *Nature photonics*, vol. 5, no. 10, p. 591, 2011.
- [22] P. Minzioni, R. Osellame, C. Sada, S. Zhao, F. Omenetto, K. B. Gylfason, T. Haraldsson, Y. Zhang, A. Ozcan, A. Wax, *et al.*, “Roadmap for optofluidics,” *Journal of Optics*, vol. 19, no. 9, p. 093003, 2017.
- [23] J. Martínez Martínez, “Transmission electron micrographs of virions from pleurochrysis carterae ccmp645 culture supernatant from laboratory experiments at the bigelow laboratory for ocean sciences, maine.” <http://lod.bco-dmo.org/id/dataset/670706>, 2016.
- [24] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [25] H. Peng, F. Long, and C. Ding, “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.
- [26] Z. Li, Z. Zhang, T. Emery, A. Scherer, and D. Psaltis, “Single mode optofluidic distributed feedback dye laser,” *Optics Express*, vol. 14, no. 2, pp. 696–701, 2006.
- [27] L. M. Lee, X. Cui, and C. Yang, “The application of on-chip optofluidic microscopy for imaging giardia lamblia trophozoites and cysts,” *Biomedical microdevices*, vol. 11, no. 5, pp. 951–958, 2009.
- [28] P. Liu, L. K. Chin, W. Ser, T. Ayi, P. Yap, T. Bourouina, and Y. Leprince-Wang, “Real-time measurement of single bacterium’s refractive index using optofluidic immersion refractometry,” *Procedia Engineering*, vol. 87, pp. 356–359, 2014.

- [29] C. Li, K. Shirahama, J. Czajkowska, M. Grzegorzek, F. Ma, and B. Zhou, “A multi-stage approach for automatic classification of environmental microorganisms,” in *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, p. 1, The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013.
- [30] C. Yang, C. Li, O. Tiebe, K. Shirahama, and M. Grzegorzek, “Shape-based classification of environmental microorganisms,” in *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pp. 3374–3379, IEEE, 2014.
- [31] H. Jiang, S. Li, W. Liu, H. Zheng, J. Liu, and Y. Zhang, “Geometry-aware cell detection with deep learning,” *Msystems*, vol. 5, no. 1, 2020.
- [32] P. Liu, L. Chin, W. Ser, T. Ayi, P. Yap, T. Bourouina, and Y. Leprince-Wang, “An optofluidic imaging system to measure the biophysical signature of single waterborne bacteria,” *Lab on a Chip*, vol. 14, no. 21, pp. 4237–4243, 2014.
- [33] S. J. Pennycook, M. Varela, C. J. Hetherington, and A. I. Kirkland, “Materials advances through aberration-corrected electron microscopy,” *MRS bulletin*, vol. 31, no. 1, pp. 36–43, 2006.
- [34] B. Mandracchia, J. Son, and S. Jia, “Super-resolution optofluidic scanning microscopy,” *Lab on a Chip*, vol. 21, no. 3, pp. 489–493, 2021.
- [35] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, “Supervised machine learning: A review of classification techniques,” *Emerging artificial intelligence applications in computer engineering*, vol. 160, no. 1, pp. 3–24, 2007.
- [36] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [37] H. Byun and S.-W. Lee, “A survey on pattern recognition applications of support vector machines,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17, no. 03, pp. 459–486, 2003.

- [38] D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance," *International journal of Remote sensing*, vol. 28, no. 5, pp. 823–870, 2007.
- [39] P. Kamavisdar, S. Saluja, and S. Agrawal, "A survey on image classification approaches and techniques," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 1, pp. 1005–1009, 2013.
- [40] S. Kosov, K. Shirahama, C. Li, and M. Grzegorzec, "Environmental microorganism classification using conditional random fields and deep convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 248–261, 2018.
- [41] M.-P. Dubuisson, A. K. Jain, and M. K. Jain, "Segmentation and classification of bacterial culture images," *Journal of Microbiological Methods*, vol. 19, no. 4, pp. 279–295, 1994.
- [42] C. Chen and X. Li, "A new wastewater bacteria classification with microscopic image analysis," in *Proceedings of the 12th WSEAS international conference on Computers*, pp. 915–921, World Scientific and Engineering Academy and Society (WSEAS), 2008.
- [43] S. Kumar and G. S. Mittal, "Rapid detection of microorganisms using image processing parameters and neural network," *Food and Bioprocess Technology*, vol. 3, no. 5, pp. 741–751, 2010.
- [44] P. Hiremath and P. Bannigidad, "Automatic identification and classification of bacilli bacterial cell growth phases," *IJCA Special Issue on Recent Trends in Image Processing and Pattern Recognition*, vol. 1, pp. 48–52, 2010.
- [45] P. Hiremath, P. Bannigidad, and S. S. Yelgond, "Identification of flagellated or fimbriated bacterial cells using digital image processing techniques," *International Journal of Computer Applications*, vol. 59, no. 12, 2012.
- [46] C. Li, K. Shirahama, M. Grzegorzec, F. Ma, and B. Zhou, "Classification of environmental microorganisms in microscopic images using shape features and support vector machines," in *2013 IEEE International Conference on Image Processing*, pp. 2435–2439, IEEE, 2013.

- [47] N. A. Mohamad, N. A. Jusoh, Z. Z. Htike, and S. L. Win, "Bacteria identification from microscopic morphology using naive bayes," *International Journal of Computer Science, Engineering and Information Technology (IJCEIT)*, vol. 4, no. 1, 2014.
- [48] G. Guo and C. R. Dyer, "Learning from examples in the small sample case: face expression recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 3, pp. 477–488, 2005.
- [49] G. Bueno, O. Deniz, A. Pedraza, J. Ruiz-Santaquiteria, J. Salido, G. Cristóbal, M. Borrego-Ramos, and S. Blanco, "Automated diatom classification (part a): hand-crafted feature approaches," *Applied Sciences*, vol. 7, no. 8, p. 753, 2017.
- [50] A. Dhindsa, S. Bhatia, S. Agrawal, and B. S. Sohi, "An improvised machine learning model based on mutual information feature selection approach for microbes classification," *Entropy*, vol. 23, no. 2, p. 257, 2021.
- [51] H. Zhang, J. E. Fritts, and S. A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," *computer vision and image understanding*, vol. 110, no. 2, pp. 260–280, 2008.
- [52] Y. Chai, V. Lempitsky, and A. Zisserman, "Bicos: A bi-level co-segmentation method for image classification," in *2011 International Conference on Computer Vision*, pp. 2579–2586, IEEE, 2011.
- [53] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman, "Tricos: A tri-level class-discriminative co-segmentation method for image classification," in *European conference on computer vision*, pp. 794–807, Springer, 2012.
- [54] N. Sanyal, A. Chatterjee, and S. Munshi, "An adaptive bacterial foraging algorithm for fuzzy entropy based image segmentation," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15489–15498, 2011.
- [55] M.-H. Horng, "Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13785–13791, 2011.

- [56] G. Kylberg, M. Uppström, K.-O. HEDLUND, G. Borgefors, and I.-M. SINTORN, “Segmentation of virus particle candidates in transmission electron microscopy images,” *Journal of microscopy*, vol. 245, no. 2, pp. 140–147, 2012.
- [57] M. Dash and H. Liu, “Feature selection for classification,” *Intelligent data analysis*, vol. 1, no. 1-4, pp. 131–156, 1997.
- [58] A. Jain and D. Zongker, “Feature selection: Evaluation, application, and small sample performance,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 19, no. 2, pp. 153–158, 1997.
- [59] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.
- [60] G. Chandrashekar and F. Sahin, “A survey on feature selection methods,” *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.
- [61] C. Ding and H. Peng, “Minimum redundancy feature selection from microarray gene expression data,” *Journal of bioinformatics and computational biology*, vol. 3, no. 02, pp. 185–205, 2005.
- [62] Y. Cai, T. Huang, L. Hu, X. Shi, L. Xie, and Y. Li, “Prediction of lysine ubiquitination with mrmr feature selection and analysis,” *Amino acids*, vol. 42, no. 4, pp. 1387–1395, 2012.
- [63] Y.-F. Gao, B.-Q. Li, Y.-D. Cai, K.-Y. Feng, Z.-D. Li, and Y. Jiang, “Prediction of active sites of enzymes by maximum relevance minimum redundancy (mrmr) feature selection,” *Molecular BioSystems*, vol. 9, no. 1, pp. 61–69, 2013.
- [64] M. Toğaçar, B. Ergen, Z. Cömert, and F. Özyurt, “A deep feature learning model for pneumonia detection applying a combination of mrmr feature selection and machine learning models,” *Irbm*, vol. 41, no. 4, pp. 212–222, 2020.
- [65] Y. Bengio, *Learning deep architectures for AI*. Now Publishers Inc, 2009.
- [66] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

- [67] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural networks*, vol. 61, pp. 85–117, 2015.
- [68] L. Deng and D. Yu, “Deep learning: methods and applications,” *Foundations and trends in signal processing*, vol. 7, no. 3–4, pp. 197–387, 2014.
- [69] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, “Handwritten digit recognition with a back-propagation network,” in *Advances in neural information processing systems*, pp. 396–404, 1990.
- [70] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [71] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 315–323, JMLR Workshop and Conference Proceedings, 2011.
- [72] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.
- [73] A. Pedraza, G. Bueno, O. Deniz, G. Cristóbal, S. Blanco, and M. Borrego-Ramos, “Automated diatom classification (part b): a deep learning approach,” *Applied Sciences*, vol. 7, no. 5, p. 460, 2017.
- [74] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [75] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.
- [76] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

- [77] C.-M. Liang, C.-C. Lai, S.-H. Wang, and Y.-H. Lin, “Environmental microorganism classification using optimized deep learning model,” *Environmental Science and Pollution Research*, pp. 1–13, 2021.
- [78] N. O’Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh, “Deep learning vs. traditional computer vision,” in *Science and Information Conference*, pp. 128–144, Springer, 2019.
- [79] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, “Deep learning for smart manufacturing: Methods and applications,” *Journal of Manufacturing Systems*, vol. 48, pp. 144–156, 2018.
- [80] J. Lu, P. Gong, J. Ye, and C. Zhang, “Learning from very few samples: A survey,” *arXiv preprint arXiv:2009.02653*, 2020.
- [81] “Public utilities board tap water quality.” <https://www.pub.gov.sg/watersupply/waterquality>. Accessed: 2021-04-28.
- [82] “Imagenet database.” <https://www.image-net.org/index.php>.
- [83] “Pascal voc (pascal visual object classes).” <https://paperswithcode.com/dataset/pascal-voc>, 2012.
- [84] W.-B. Chen and C. Zhang, “An automated bacterial colony counting and classification system,” *Information Systems Frontiers*, vol. 11, no. 4, pp. 349–368, 2009.
- [85] M. K. Gilbert, C. Frick, A. Wodowski, and F. Vogt, “Spectroscopic imaging for detection and discrimination of different e. coli strains,” *Applied spectroscopy*, vol. 63, no. 1, pp. 6–13, 2009.
- [86] W. M. Ahmed, B. Bayraktar, A. K. Bhunia, E. D. Hirleman, J. P. Robinson, and B. Rajwa, “Classification of bacterial contamination using image processing and distributed computing,” *IEEE journal of biomedical and health informatics*, vol. 17, no. 1, pp. 232–239, 2012.
- [87] S. Ayas and M. Ekinici, “Random forest-based tuberculosis bacteria classification in images of zn-stained sputum smear samples,” *Signal, Image and Video Processing*, vol. 8, no. 1, pp. 49–61, 2014.

- [88] Y. Mao, H. Liu, R. Ye, Y. Shi, and Z. Song, "Detection and segmentation of virus plaque using hog and svm: toward automatic plaque assay," *Bio-medical materials and engineering*, vol. 24, no. 6, pp. 3187–3198, 2014.
- [89] K. Zhang, M. Fei, X. Li, and H. Zhou, "Adaptive bacteria colony picking in unstructured environments using intensity histogram and unascertained ls-svm classifier," *The Scientific World Journal*, vol. 2014, 2014.
- [90] "Science photo library (spl)." <https://www.denniskunkel.com/index.html>.
- [91] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.
- [92] M. M. Galloway, "Texture analysis using grey level run lengths," *NASA STI/Recon Technical Report N*, vol. 75, p. 18555, 1974.
- [93] I. T. Jolliffe, "Principal component analysis: a beginner's guide—i. introduction and application," *Weather*, vol. 45, no. 10, pp. 375–382, 1990.
- [94] I. T. Jolliffe, *Principal component analysis*. Springer, 2002.
- [95] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.
- [96] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, pp. 556–562, 2001.
- [97] R. K. McConnell, "Method of and apparatus for pattern recognition," Jan. 28 1986. US Patent 4,567,610.
- [98] W. T. Freeman and M. Roth, "Orientation histograms for hand gesture recognition," in *International workshop on automatic face and gesture recognition*, vol. 12, pp. 296–301, 1995.
- [99] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886–893, Ieee, 2005.

- [100] P. Mitra, C. Murthy, and S. K. Pal, “Unsupervised feature selection using feature similarity,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 3, pp. 301–312, 2002.
- [101] J. Sauvola and M. Pietikäinen, “Adaptive document image binarization,” *Pattern recognition*, vol. 33, no. 2, pp. 225–236, 2000.
- [102] P. Jana, S. Ghosh, S. K. Bera, and R. Sarkar, “Handwritten document image binarization: An adaptive k-means based approach,” in *2017 IEEE Calcutta Conference (CALCON)*, pp. 226–230, IEEE, 2017.
- [103] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.
- [104] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.
- [105] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, *et al.*, “Searching for mobilenetv3,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1314–1324, 2019.
- [106] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.
- [107] T. Falk, D. Mai, R. Bensch, Ö. Çiçek, A. Abdulkadir, Y. Marrakchi, A. Böhm, J. Deubner, Z. Jäckel, K. Seiwald, *et al.*, “U-net: deep learning for cell counting, detection, and morphometry,” *Nature methods*, vol. 16, no. 1, pp. 67–70, 2019.
- [108] P. Schelb, S. Kohl, J. P. Radtke, M. Wiesenfarth, P. Kickingeder, S. Bickelhaupt, T. A. Kuder, A. Stenzinger, M. Hohenfellner, H.-P. Schlemmer, *et al.*, “Classification of cancer at prostate mri: deep learning versus clinical pi-rads assessment,” *Radiology*, vol. 293, no. 3, pp. 607–617, 2019.

- [109] Z. Li, C. Li, Y. Yao, J. Zhang, M. M. Rahaman, H. Xu, F. Kulwa, B. Lu, X. Zhu, and T. Jiang, “Emds-5: Environmental microorganism image dataset fifth version for multiple image analysis tasks,” *Plos one*, vol. 16, no. 5, p. e0250631, 2021.
- [110] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Deepface: Closing the gap to human-level performance in face verification,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1701–1708, 2014.
- [111] H. Zhu, S. Mavandadi, A. F. Coskun, O. Yaglidere, and A. Ozcan, “Optofluidic fluorescent imaging cytometry on a cell phone,” *Analytical chemistry*, vol. 83, no. 17, pp. 6641–6647, 2011.
- [112] H. Alshamsi, V. Kepuska, and H. Meng, “Automated facial expression recognition app development on smart phones using cloud computing,” in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pp. 577–583, IEEE, 2017.
- [113] M. A. Kadampur and S. Al Riyae, “Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images,” *Informatics in Medicine Unlocked*, vol. 18, p. 100282, 2020.
- [114] J. F. Zimmerman and B. Tian, “Multifunctional optofluidic brain probes,” *Nature biomedical engineering*, vol. 3, no. 8, pp. 596–597, 2019.
- [115] S. Kumar and G. S. Mittal, “Geometric and optical characteristics of five microorganisms for rapid detection using image processing,” *Biosystems engineering*, vol. 99, no. 1, pp. 1–8, 2008.
- [116] S. Kumar and G. S. Mittal, “Textural characteristics of five microorganisms for rapid detection using image processing,” *Journal of food process engineering*, vol. 32, no. 1, pp. 126–143, 2009.
- [117] Y. Kanuki, N. Ohta, and A. Nagai, “Automatic compensation of radial distortion by minimizing entropy of histogram of oriented gradients,” in *2013 2nd IAPR Asian Conference on Pattern Recognition*, pp. 912–916, IEEE, 2013.

- [118] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, “Pruning convolutional neural networks for resource efficient inference,” in *5th International Conference on Learning Representations, ICLR 2017-Conference Track Proceedings*, 2019.
- [119] “Python code to count flops in pytorch framework.” <https://github.com/sovrasov/flops-counter.pytorch>. Accessed: 2021-05-04.
- [120] “Matlab code to count flops.” <https://www.mathworks.com/matlabcentral/fileexchange/50608-counting-the-floating-point-operations-flops>. Accessed: 2021-05-04.

Appendix A

Appendix A - Boxplot Check on Features in Kumar's Method

S. Kumar and G.S. Mittal designed a classification algorithm for their microscopic bacterial images [43],[115],[116]. This algorithm used 9 features with the artificial neural network (ANN) as the classifier and achieved 100% on their dataset [43]. There were 2 morphological features (width, shape factor), 3 optical features (mean value of the intensity histogram, 10 percentile value of the intensity histogram, and 99 percentile value of the intensity histogram), and 4 textual features (horizontal run length non-uniformity, 45° run length non-uniformity, entropy of the of co-occurrence matrix, and sum entropy of the of co-occurrence matrix).

The following figures (Figures A.1, A.2) show some illustrations of the boxplot of these features extracted from our dataset of optofluidic protozoan images. Here, "MinorL" refers to the minor axis length, i.e., the width; "Round" refers to the roundness, i.e., the shape factor; "Mean", "10Prctile", and "99Prctile" refer to the mean, 10 percentile, and 99 percentile value of the intensity histogram respectively; "HoriRLNU" refers to the horizontal run length non-uniformity (RLNU); "45RLNU" refers to the 45° RLNU; "ECoMat", and "SECoMat" refer to the entropy and sum entropy of the of co-occurrence matrix respectively. "G", "C", and "O" refer to the class G, class C, and class O of our dataset see (Section 3.1). It can be easily seen that all features did not pass the boxplot check (Section 4.2.2) in Figure A.1, meaning that these features, which were picked and performed well for S. Kumar and G.S. Mittal's microscopic bacterial image classification, did not have a satisfying ability to classify our optofluidic images. This was also proved in the classification results (Section 4.4).

FIGURE A.1: Illustration of boxplot of the features in Kumar's paper [43] using the images in our dataset

Training images of 1 fold displayed.

FIGURE A.2: Illustration of boxplot of the features in Kumar's paper [43] using the images in our dataset

Testing images of 1 fold displayed.

The details of the run length matrix and co-occurrence matrix are explained below.

For an image, its run length matrix, $p_\theta(i, k)$, shows the number of runs (i.e. consecutive collinear points) with angle θ , grey level i , and run length k . Usually, θ can be 0° (i.e. horizontal), 45° , 90° , and 135° . One example for run length matrix is shown in Figure A.3. In Kumar's method[43], the 0° , and 45° run length matrix are required.

FIGURE A.3: Run length matrix example

The run length non-uniformity (RLN) is calculated using Equation A.1.

$$RLN = \frac{\sum_{k=1}^{N_l} [\sum_{i=1}^{N_g} p_\theta(i, k)]^2}{\sum_{i=1}^{N_g} \sum_{k=1}^{N_l} p_\theta(i, k)} \quad (A.1)$$

Here, N_l denotes the longest possible run length, and N_g denotes the number of grey levels.

For an image, its grey level co-occurrence matrix (GLCM), $g_{\Delta x, \Delta y}(i, j)$, shows the number of pixel pairs with grey levels i and j respectively, and position offset $[\Delta x, \Delta y]$. One example of GLCM is shown in Figure A.4. In Kumar's method[43], the $[0, 1]$ (1-pixel distance, horizontal direction) offset is used to generate the GLCM for entropy and sum entropy calculation.

FIGURE A.4: Grey level co-occurrence matrix example

The entropy of GLCM is calculated using Equation A.2.

$$Entropy(GLCM) = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} g_{\Delta x, \Delta y}(i, j) \log_2[g_{\Delta x, \Delta y}(i, j)] \quad (A.2)$$

The sum entropy of GLCM is calculated using Equation A.3 and A.4.

$$g_{plus}(m) = \sum_{i=1}^{N_g} \sum_{\substack{j=1, \\ i+j=m}}^{N_g} [g_{\Delta x, \Delta y}(i, j)] \quad (A.3)$$

$$SumEntropy(GLCM) = - \sum_{m=2}^{2N_g} g_{plus}(m) \log_2[g_{plus}(m)] \quad (A.4)$$

Appendix B

Appendix B - Boxplot Check on Features in Bueno's Method

G. Bueno, O. Deniz, et al. [49] extracted 1460 existing handcrafted features on their diatom image dataset. In their paper, the aspect ratio (AR) (introduced in Section 4.2.1) was defined as eccentricity 1, the ratio of the maximum and minimum distances between the object centroid and its boundary was defined as eccentricity 2, the eccentricity (introduced in Section 4.2.1) was defined as eccentricity 3.

We did the boxplot check (explained in Section 4.2.2) of these features on our waterborne pathogen images. In 10-fold cross-validation, the final boxplot check results were then averaged and rounded (to 1 or 0) value of the results of 10 folds' training sets. It can be observed from Table B.1 that AR is the only morphological feature that passed the boxplot check under all cases.

TABLE B.1: Boxplot check results of the features in Bueno's paper [49] extracted using the images in our dataset

For the original dataset, numbers of G, C, O class images are 147, 117, 148 respectively. For the augmented datasets, original images were rotated to generate more than 1000 images and then randomly selected certain numbers. 1 refers to passing the boxplot check, 0 refers to not passing the boxplot check. # refers to feature number. AND refers to the AND result of all the 7 cases leftward.

			Number of images per class									
#	Category	Name	147, 117, 148	30	70	100	300	700	1000	AND		
		174										

1	Morphological	Area	1	0	0	0	0	0	0	0
2	Morphological	Eccentricity 1	1	1	1	1	1	1	1	1
3	Morphological	Eccentricity 2	0	1	1	1	1	1	1	0
4	Morphological	Eccentricity 3	0	1	1	1	1	1	1	0
5	Morphological	Perimeter	0	1	0	1	1	1	1	0
6	Morphological	Compactness	0	1	0	0	0	0	0	0
7	Morphological	Fullness	0	1	1	1	1	1	1	0
8	Statistical	Mean	0	0	0	0	0	0	0	0
9	Statistical	Mode	1	1	1	1	1	1	1	1
10	Statistical	Min	1	1	1	1	1	1	1	1
11	Statistical	Max	1	1	1	1	1	1	1	1
12	Statistical	Variance	1	1	1	1	1	1	1	1
13	Statistical	Range	1	1	1	1	1	1	1	1
14	Statistical	Entropy	0	0	0	0	0	0	0	0
15	Statistical	1st Quartile	1	1	1	1	1	1	1	1
16	Statistical	2nd Quartile	1	1	1	1	1	1	1	1
17	Statistical	3rd Quartile	0	0	0	0	0	0	0	0
18	Statistical	Interquartile Range	0	0	0	0	0	0	0	0
19	Statistical	Asymmetry	1	1	1	1	1	1	1	1
20	Statistical	Kurtosis	1	1	1	1	1	1	1	1
21	Statistical	Energy (d = 1, 0°)	1	0	0	0	0	0	0	0
22	Statistical	Variance (d = 1, 0°)	0	0	0	0	0	0	0	0
23	Statistical	Contrast (d = 1, 0°)	0	0	0	0	0	0	0	0
24	Statistical	Dissimilarity (d = 1, 0°)	0	0	0	0	0	0	0	0
25	Statistical	Correlation (d = 1, 0°)	0	1	1	1	1	1	1	0
26	Statistical	Autocorrelation (d = 1, 0°)	1	0	0	0	0	0	0	0
27	Statistical	Entropy (d = 1, 0°)	0	0	0	0	0	0	0	0
28	Statistical	Measure of Correlation 1 (d = 1, 0°)	0	0	0	0	0	0	0	0
29	Statistical	Measure of Correlation 2 (d = 1, 0°)	1	1	1	1	1	1	1	1

30	Statistical	Cluster Shade (d = 1, 0°)	1	0	0	0	0	0	0	0
31	Statistical	Cluster Prominence (d = 1, 0°)	1	0	0	0	0	0	0	0
32	Statistical	Max Probability (d = 1, 0°)	1	0	0	0	0	0	0	0
33	Statistical	Sum Average (d = 1, 0°)	1	0	0	0	0	0	0	0
34	Statistical	Sum Entropy (d = 1, 0°)	1	0	0	0	0	0	0	0
35	Statistical	Sum Variance (d = 1, 0°)	1	0	0	0	0	0	0	0
36	Statistical	Difference Entropy (d = 1, 0°)	0	0	0	0	0	0	0	0
37	Statistical	Difference Variance (d = 1, 0°)	0	0	0	0	0	0	0	0
38	Statistical	Homogeneity 1 (d = 1, 0°)	0	0	0	0	0	0	0	0
39	Statistical	Homogeneity 2 (d = 1, 0°)	0	0	0	0	0	0	0	0
40	Statistical	Energy (d = 1, 45°)	1	0	0	0	0	0	0	0
41	Statistical	Variance (d = 1, 45°)	0	0	0	0	0	0	0	0
42	Statistical	Contrast (d = 1, 45°)	0	1	1	1	1	1	1	0
43	Statistical	Dissimilarity (d = 1, 45°)	0	0	0	0	0	0	0	0
44	Statistical	Correlation (d = 1, 45°)	0	1	1	1	1	1	1	0
45	Statistical	Autocorrelation (d = 1, 45°)	1	0	0	0	0	0	0	0
46	Statistical	Entropy (d = 1, 45°)	0	0	0	0	0	0	0	0
47	Statistical	Measure of Correlation 1 (d = 1, 45°)	0	0	0	0	0	0	0	0
48	Statistical	Measure of Correlation 2 (d = 1, 45°)	1	1	1	1	1	1	1	1
49	Statistical	Cluster Shade (d = 1, 45°)	1	0	0	0	0	0	0	0
50	Statistical	Cluster Prominence (d = 1, 45°)	1	0	0	0	0	0	0	0
51	Statistical	Max Probability (d = 1, 45°)	1	0	0	0	0	0	0	0
52	Statistical	Sum Average (d = 1, 45°)	1	0	0	0	0	0	0	0
53	Statistical	Sum Entropy (d = 1, 45°)	1	0	0	0	0	0	0	0
54	Statistical	Sum Variance (d = 1, 45°)	1	0	0	0	0	0	0	0
55	Statistical	Difference Entropy (d = 1, 45°)	0	0	0	0	0	0	0	0
56	Statistical	Difference Variance (d = 1, 45°)	0	1	1	1	1	1	1	0
57	Statistical	Homogeneity 1 (d = 1, 45°)	0	0	0	0	0	0	0	0
58	Statistical	Homogeneity 2 (d = 1, 45°)	0	0	0	0	0	0	0	0
59	Statistical	Energy (d = 1, 90°)	1	0	0	0	0	0	0	0

60	Statistical	Variance (d = 1, 90°)	0	0	0	0	0	0	0	0
61	Statistical	Contrast (d = 1, 90°)	0	1	1	1	1	1	1	0
62	Statistical	Dissimilarity (d = 1, 90°)	0	0	1	1	1	1	1	0
63	Statistical	Correlation (d = 1, 90°)	0	1	1	1	1	1	1	0
64	Statistical	Autocorrelation (d = 1, 90°)	1	0	0	0	0	0	0	0
65	Statistical	Entropy (d = 1, 90°)	1	0	0	0	0	0	0	0
66	Statistical	Measure of Correlation 1 (d = 1, 90°)	0	0	0	0	0	0	0	0
67	Statistical	Measure of Correlation 2 (d = 1, 90°)	1	1	1	1	1	1	1	1
68	Statistical	Cluster Shade (d = 1, 90°)	1	0	0	0	0	0	0	0
69	Statistical	Cluster Prominence (d = 1, 90°)	1	0	0	0	0	0	0	0
70	Statistical	Max Probability (d = 1, 90°)	1	0	0	0	0	0	0	0
71	Statistical	Sum Average (d = 1, 90°)	1	0	0	0	0	0	0	0
72	Statistical	Sum Entropy (d = 1, 90°)	1	0	0	0	0	0	0	0
73	Statistical	Sum Variance (d = 1, 90°)	1	0	0	0	0	0	0	0
74	Statistical	Difference Entropy (d = 1, 90°)	0	0	0	0	0	0	0	0
75	Statistical	Difference Variance (d = 1, 90°)	0	1	1	1	1	1	1	0
76	Statistical	Homogeneity 1 (d = 1, 90°)	0	0	0	0	0	0	0	0
77	Statistical	Homogeneity 2 (d = 1, 90°)	0	0	0	0	0	0	0	0
78	Statistical	Energy (d = 1, 135°)	1	0	0	0	0	0	0	0
79	Statistical	Variance (d = 1, 135°)	0	0	0	0	0	0	0	0
80	Statistical	Contrast (d = 1, 135°)	0	1	1	1	1	1	1	0
81	Statistical	Dissimilarity (d = 1, 135°)	0	0	1	0	0	0	0	0
82	Statistical	Correlation (d = 1, 135°)	0	1	1	1	1	1	1	0
83	Statistical	Autocorrelation (d = 1, 135°)	1	0	0	0	0	0	0	0
84	Statistical	Entropy (d = 1, 135°)	0	0	0	0	0	0	0	0
85	Statistical	Measure of Correlation 1 (d = 1, 135°)	0	0	0	0	0	0	0	0
86	Statistical	Measure of Correlation 2 (d = 1, 135°)	1	1	1	1	1	1	1	1
87	Statistical	Cluster Shade (d = 1, 135°)	1	0	0	0	0	0	0	0
88	Statistical	Cluster Prominence (d = 1, 135°)	1	0	0	0	0	0	0	0
89	Statistical	Max Probability (d = 1, 135°)	1	0	0	0	0	0	0	0

90	Statistical	Sum Average (d = 1, 135°)	1	0	0	0	0	0	0	0
91	Statistical	Sum Entropy (d = 1, 135°)	1	0	0	0	0	0	0	0
92	Statistical	Sum Variance (d = 1, 135°)	1	0	0	0	0	0	0	0
93	Statistical	Difference Entropy (d = 1, 135°)	0	0	0	0	0	0	0	0
94	Statistical	Difference Variance (d = 1, 135°)	0	1	1	1	1	1	1	0
95	Statistical	Homogeneity 1 (d = 1, 135°)	0	0	0	0	0	0	0	0
96	Statistical	Homogeneity 2 (d = 1, 135°)	0	0	0	0	0	0	0	0
97	Statistical	Energy (d = 3, 0°)	0	0	0	0	0	0	0	0
98	Statistical	Variance (d = 3, 0°)	0	0	0	0	0	0	0	0
99	Statistical	Contrast (d = 3, 0°)	0	0	1	1	1	1	1	0
100	Statistical	Dissimilarity (d = 3, 0°)	0	0	0	0	0	0	0	0
101	Statistical	Correlation (d = 3, 0°)	0	1	1	1	1	1	1	0
102	Statistical	Autocorrelation (d = 3, 0°)	1	0	0	0	0	0	0	0
103	Statistical	Entropy (d = 3, 0°)	0	0	0	0	0	0	0	0
104	Statistical	Measure of Correlation 1 (d = 3, 0°)	0	0	0	0	0	0	0	0
105	Statistical	Measure of Correlation 2 (d = 3, 0°)	1	1	1	1	1	1	1	1
106	Statistical	Cluster Shade (d = 3, 0°)	1	0	0	0	0	0	0	0
107	Statistical	Cluster Prominence (d = 3, 0°)	1	0	0	0	0	0	0	0
108	Statistical	Max Probability (d = 3, 0°)	1	0	0	0	0	0	0	0
109	Statistical	Sum Average (d = 3, 0°)	1	0	0	0	0	0	0	0
110	Statistical	Sum Entropy (d = 3, 0°)	1	0	0	0	0	0	0	0
111	Statistical	Sum Variance (d = 3, 0°)	1	0	0	0	0	0	0	0
112	Statistical	Difference Entropy (d = 3, 0°)	0	0	0	0	0	0	0	0
113	Statistical	Difference Variance (d = 3, 0°)	0	0	1	0	0	0	0	0
114	Statistical	Homogeneity 1 (d = 3, 0°)	0	0	0	0	0	0	0	0
115	Statistical	Homogeneity 2 (d = 3, 0°)	0	0	0	0	0	0	0	0
116	Statistical	Energy (d = 3, 45°)	0	0	0	0	0	0	0	0
117	Statistical	Variance (d = 3, 45°)	0	0	0	0	0	0	0	0
118	Statistical	Contrast (d = 3, 45°)	0	1	1	1	1	1	1	0
119	Statistical	Dissimilarity (d = 3, 45°)	0	0	1	0	1	0	0	0

120	Statistical	Correlation (d = 3, 45°)	0	1	1	1	1	1	1	0
121	Statistical	Autocorrelation (d = 3, 45°)	1	0	0	0	0	0	0	0
122	Statistical	Entropy (d = 3, 45°)	0	0	0	0	0	0	0	0
123	Statistical	Measure of Correlation 1 (d = 3, 45°)	0	0	0	0	0	0	0	0
124	Statistical	Measure of Correlation 2 (d = 3, 45°)	1	1	1	1	1	1	1	1
125	Statistical	Cluster Shade (d = 3, 45°)	1	0	0	0	0	0	0	0
126	Statistical	Cluster Prominence (d = 3, 45°)	1	0	0	0	0	0	0	0
127	Statistical	Max Probability (d = 3, 45°)	1	0	0	0	0	0	0	0
128	Statistical	Sum Average (d = 3, 45°)	1	0	0	0	0	0	0	0
129	Statistical	Sum Entropy (d = 3, 45°)	1	0	0	0	0	0	0	0
130	Statistical	Sum Variance (d = 3, 45°)	1	0	0	0	0	0	0	0
131	Statistical	Difference Entropy (d = 3, 45°)	0	0	0	0	0	0	0	0
132	Statistical	Difference Variance (d = 3, 45°)	0	0	1	1	1	1	1	0
133	Statistical	Homogeneity 1 (d = 3, 45°)	0	0	0	0	0	0	0	0
134	Statistical	Homogeneity 2 (d = 3, 45°)	0	0	0	0	0	0	0	0
135	Statistical	Energy (d = 3, 90°)	1	0	0	0	0	0	0	0
136	Statistical	Variance (d = 3, 90°)	0	0	0	0	0	0	0	0
137	Statistical	Contrast (d = 3, 90°)	0	1	1	1	1	1	1	0
138	Statistical	Dissimilarity (d = 3, 90°)	0	0	1	0	1	0	0	0
139	Statistical	Correlation (d = 3, 90°)	0	1	1	1	1	1	1	0
140	Statistical	Autocorrelation (d = 3, 90°)	1	0	0	0	0	0	0	0
141	Statistical	Entropy (d = 3, 90°)	0	0	0	0	0	0	0	0
142	Statistical	Measure of Correlation 1 (d = 3, 90°)	0	0	0	0	0	0	0	0
143	Statistical	Measure of Correlation 2 (d = 3, 90°)	1	1	1	1	1	1	1	1
144	Statistical	Cluster Shade (d = 3, 90°)	1	0	0	0	0	0	0	0
145	Statistical	Cluster Prominence (d = 3, 90°)	1	0	0	0	0	0	0	0
146	Statistical	Max Probability (d = 3, 90°)	1	0	0	0	0	0	0	0
147	Statistical	Sum Average (d = 3, 90°)	1	0	0	0	0	0	0	0
148	Statistical	Sum Entropy (d = 3, 90°)	1	0	0	0	0	0	0	0
149	Statistical	Sum Variance (d = 3, 90°)	1	0	0	0	0	0	0	0

150	Statistical	Difference Entropy (d = 3, 90°)	0	0	1	0	0	0	0	0
151	Statistical	Difference Variance (d = 3, 90°)	0	0	1	0	1	1	1	0
152	Statistical	Homogeneity 1 (d = 3, 90°)	0	0	1	0	0	0	0	0
153	Statistical	Homogeneity 2 (d = 3, 90°)	0	0	0	0	0	0	0	0
154	Statistical	Energy (d = 3, 135°)	0	0	0	0	0	0	0	0
155	Statistical	Variance (d = 3, 135°)	0	0	0	0	0	0	0	0
156	Statistical	Contrast (d = 3, 135°)	0	1	1	1	1	1	1	0
157	Statistical	Dissimilarity (d = 3, 135°)	0	0	1	1	1	1	1	0
158	Statistical	Correlation (d = 3, 135°)	0	0	1	1	1	1	1	0
159	Statistical	Autocorrelation (d = 3, 135°)	1	0	0	0	0	0	0	0
160	Statistical	Entropy (d = 3, 135°)	0	0	0	0	0	0	0	0
161	Statistical	Measure of Correlation 1 (d = 3, 135°)	0	0	0	0	0	0	0	0
162	Statistical	Measure of Correlation 2 (d = 3, 135°)	1	1	1	1	1	1	1	1
163	Statistical	Cluster Shade (d = 3, 135°)	1	0	0	0	0	0	0	0
164	Statistical	Cluster Prominence (d = 3, 135°)	1	0	0	0	0	0	0	0
165	Statistical	Max Probability (d = 3, 135°)	1	0	0	0	0	0	0	0
166	Statistical	Sum Average (d = 3, 135°)	1	0	0	0	0	0	0	0
167	Statistical	Sum Entropy (d = 3, 135°)	1	0	0	0	0	0	0	0
168	Statistical	Sum Variance (d = 3, 135°)	1	0	0	0	0	0	0	0
169	Statistical	Difference Entropy (d = 3, 135°)	0	0	0	0	1	0	0	0
170	Statistical	Difference Variance (d = 3, 135°)	0	0	1	1	1	1	1	0
171	Statistical	Homogeneity 1 (d = 3, 135°)	0	0	0	0	1	0	0	0
172	Statistical	Homogeneity 2 (d = 3, 135°)	0	0	0	0	0	0	0	0
173	Statistical	Energy (d = 5, 0°)	0	0	0	0	0	0	0	0
174	Statistical	Variance (d = 5, 0°)	0	0	0	0	0	0	0	0
175	Statistical	Contrast (d = 5, 0°)	0	1	1	1	1	1	1	0
176	Statistical	Dissimilarity (d = 5, 0°)	0	0	0	0	1	0	0	0
177	Statistical	Correlation (d = 5, 0°)	0	0	1	1	1	1	1	0
178	Statistical	Autocorrelation (d = 5, 0°)	1	0	0	0	0	0	0	0
179	Statistical	Entropy (d = 5, 0°)	0	0	0	0	0	0	0	0

180	Statistical	Measure of Correlation 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
181	Statistical	Measure of Correlation 2 (d = 5, 0°)	1	1	1	1	1	1	1	1
182	Statistical	Cluster Shade (d = 5, 0°)	1	0	0	0	0	0	0	0
183	Statistical	Cluster Prominence (d = 5, 0°)	1	0	0	0	0	0	0	0
184	Statistical	Max Probability (d = 5, 0°)	1	0	0	0	0	0	0	0
185	Statistical	Sum Average (d = 5, 0°)	1	0	0	0	0	0	0	0
186	Statistical	Sum Entropy (d = 5, 0°)	1	0	0	0	0	0	0	0
187	Statistical	Sum Variance (d = 5, 0°)	1	0	0	0	0	0	0	0
188	Statistical	Difference Entropy (d = 5, 0°)	0	0	0	0	0	0	0	0
189	Statistical	Difference Variance (d = 5, 0°)	0	0	1	1	1	1	1	0
190	Statistical	Homogeneity 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
191	Statistical	Homogeneity 2 (d = 5, 0°)	0	0	0	0	0	0	0	0
192	Statistical	Energy (d = 5, 45°)	0	0	0	0	0	0	0	0
193	Statistical	Variance (d = 5, 45°)	0	0	0	0	0	0	0	0
194	Statistical	Contrast (d = 5, 45°)	0	1	1	1	1	1	1	0
195	Statistical	Dissimilarity (d = 5, 45°)	0	0	1	1	1	1	1	0
196	Statistical	Correlation (d = 5, 45°)	0	0	0	0	0	0	0	0
197	Statistical	Autocorrelation (d = 5, 45°)	1	0	0	0	0	0	0	0
198	Statistical	Entropy (d = 5, 45°)	0	0	0	0	0	0	0	0
199	Statistical	Measure of Correlation 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
200	Statistical	Measure of Correlation 2 (d = 5, 45°)	1	1	1	1	1	1	1	1
201	Statistical	Cluster Shade (d = 5, 45°)	1	0	0	0	0	0	0	0
202	Statistical	Cluster Prominence (d = 5, 45°)	1	0	0	0	0	0	0	0
203	Statistical	Max Probability (d = 5, 45°)	1	0	0	0	0	0	0	0
204	Statistical	Sum Average (d = 5, 45°)	1	0	0	0	0	0	0	0
205	Statistical	Sum Entropy (d = 5, 45°)	1	0	0	0	0	0	0	0
206	Statistical	Sum Variance (d = 5, 45°)	1	0	0	0	0	0	0	0
207	Statistical	Difference Entropy (d = 5, 45°)	0	0	0	0	0	0	0	0
208	Statistical	Difference Variance (d = 5, 45°)	0	1	1	1	1	1	1	0
209	Statistical	Homogeneity 1 (d = 5, 45°)	0	0	0	0	0	0	0	0

210	Statistical	Homogeneity 2 (d = 5, 45°)	0	0	0	0	0	0	0	0
211	Statistical	Energy (d = 5, 90°)	0	0	0	0	0	0	0	0
212	Statistical	Variance (d = 5, 90°)	0	0	0	0	0	0	0	0
213	Statistical	Contrast (d = 5, 90°)	0	1	1	1	1	1	1	0
214	Statistical	Dissimilarity (d = 5, 90°)	0	0	1	1	1	1	1	0
215	Statistical	Correlation (d = 5, 90°)	0	0	0	0	0	0	0	0
216	Statistical	Autocorrelation (d = 5, 90°)	1	0	0	0	0	0	0	0
217	Statistical	Entropy (d = 5, 90°)	0	0	0	0	0	0	0	0
218	Statistical	Measure of Correlation 1 (d = 5, 90°)	0	0	0	0	0	0	0	0
219	Statistical	Measure of Correlation 2 (d = 5, 90°)	1	1	1	1	1	1	1	1
220	Statistical	Cluster Shade (d = 5, 90°)	1	0	0	0	0	0	0	0
221	Statistical	Cluster Prominence (d = 5, 90°)	1	0	0	0	0	0	0	0
222	Statistical	Max Probability (d = 5, 90°)	1	0	0	0	0	0	0	0
223	Statistical	Sum Average (d = 5, 90°)	1	0	0	0	0	0	0	0
224	Statistical	Sum Entropy (d = 5, 90°)	1	0	0	0	0	0	0	0
225	Statistical	Sum Variance (d = 5, 90°)	1	0	0	0	0	0	0	0
226	Statistical	Difference Entropy (d = 5, 90°)	0	0	1	1	0	0	0	0
227	Statistical	Difference Variance (d = 5, 90°)	0	1	1	1	1	1	1	0
228	Statistical	Homogeneity 1 (d = 5, 90°)	0	0	1	1	0	0	0	0
229	Statistical	Homogeneity 2 (d = 5, 90°)	0	0	0	1	0	0	0	0
230	Statistical	Energy (d = 5, 135°)	0	0	0	0	0	0	0	0
231	Statistical	Variance (d = 5, 135°)	0	0	0	0	0	0	0	0
232	Statistical	Contrast (d = 5, 135°)	0	1	1	1	1	1	1	0
233	Statistical	Dissimilarity (d = 5, 135°)	0	0	1	1	1	1	1	0
234	Statistical	Correlation (d = 5, 135°)	0	0	0	0	0	0	0	0
235	Statistical	Autocorrelation (d = 5, 135°)	1	0	0	0	0	0	0	0
236	Statistical	Entropy (d = 5, 135°)	0	0	0	0	0	0	0	0
237	Statistical	Measure of Correlation 1 (d = 5, 135°)	0	0	0	1	0	0	0	0
238	Statistical	Measure of Correlation 2 (d = 5, 135°)	1	1	1	1	1	1	1	1
239	Statistical	Cluster Shade (d = 5, 135°)	1	0	0	0	0	0	0	0

240	Statistical	Cluster Prominence (d = 5, 135°)	1	0	0	0	0	0	0	0
241	Statistical	Max Probability (d = 5, 135°)	1	0	0	0	0	0	0	0
242	Statistical	Sum Average (d = 5, 135°)	1	0	0	0	0	0	0	0
243	Statistical	Sum Entropy (d = 5, 135°)	1	0	0	0	0	0	0	0
244	Statistical	Sum Variance (d = 5, 135°)	1	0	0	0	0	0	0	0
245	Statistical	Difference Entropy (d = 5, 135°)	0	0	0	1	1	1	1	0
246	Statistical	Difference Variance (d = 5, 135°)	0	1	1	1	1	1	1	0
247	Statistical	Homogeneity 1 (d = 5, 135°)	0	0	0	1	1	1	1	0
248	Statistical	Homogeneity 2 (d = 5, 135°)	0	0	0	1	1	0	0	0
249	LBP Statistical	Mean	1	1	1	1	1	1	1	1
250	LBP Statistical	Mode	1	1	1	1	1	1	1	1
251	LBP Statistical	Min	1	1	1	1	1	1	1	1
252	LBP Statistical	Max	1	1	1	1	1	1	1	1
253	LBP Statistical	Variance	1	1	1	1	1	1	1	1
254	LBP Statistical	Range	1	1	1	1	1	1	1	1
255	LBP Statistical	Entropy	1	1	1	1	1	1	1	1
256	LBP Statistical	1st Quartile	0	0	0	0	0	0	0	0
257	LBP Statistical	2nd Quartile	0	0	0	0	0	0	0	0
258	LBP Statistical	3rd Quartile	0	0	0	0	0	0	0	0
259	LBP Statistical	Interquartile Range	0	0	0	0	0	0	0	0
260	LBP Statistical	Asymmetry	1	1	1	1	1	1	1	1
261	LBP Statistical	Kurtosis	1	1	1	1	1	1	1	1
262	LBP Statistical	Energy (d = 1, 0°)	1	0	1	0	0	0	0	0
263	LBP Statistical	Variance (d = 1, 0°)	1	1	1	1	1	1	1	1
264	LBP Statistical	Contrast (d = 1, 0°)	1	0	0	0	0	0	0	0
265	LBP Statistical	Dissimilarity (d = 1, 0°)	1	0	0	0	0	0	0	0
266	LBP Statistical	Correlation (d = 1, 0°)	1	1	1	1	1	1	1	1
267	LBP Statistical	Autocorrelation (d = 1, 0°)	1	0	1	0	0	0	0	0
268	LBP Statistical	Entropy (d = 1, 0°)	1	0	1	0	0	0	0	0
269	LBP Statistical	Measure of Correlation 1 (d = 1, 0°)	0	0	0	0	0	0	0	0

270	LBP Statistical	Measure of Correlation 2 (d = 1, 0°)	1	1	1	1	1	1	1	1
271	LBP Statistical	Cluster Shade (d = 1, 0°)	1	1	1	1	1	1	1	1
272	LBP Statistical	Cluster Prominence (d = 1, 0°)	1	1	1	1	1	1	1	1
273	LBP Statistical	Max Probability (d = 1, 0°)	1	0	1	0	0	0	0	0
274	LBP Statistical	Sum Average (d = 1, 0°)	1	1	1	1	1	1	1	1
275	LBP Statistical	Sum Entropy (d = 1, 0°)	1	0	1	0	1	0	0	0
276	LBP Statistical	Sum Variance (d = 1, 0°)	1	0	1	0	1	0	0	0
277	LBP Statistical	Difference Entropy (d = 1, 0°)	0	0	0	0	0	0	0	0
278	LBP Statistical	Difference Variance (d = 1, 0°)	1	0	0	0	0	0	0	0
279	LBP Statistical	Homogeneity 1 (d = 1, 0°)	0	0	0	0	0	0	0	0
280	LBP Statistical	Homogeneity 2 (d = 1, 0°)	0	0	0	0	0	0	0	0
281	LBP Statistical	Energy (d = 1, 45°)	1	0	1	0	0	0	0	0
282	LBP Statistical	Variance (d = 1, 45°)	1	1	1	1	1	1	1	1
283	LBP Statistical	Contrast (d = 1, 45°)	1	0	0	0	0	0	0	0
284	LBP Statistical	Dissimilarity (d = 1, 45°)	1	0	0	0	0	0	0	0
285	LBP Statistical	Correlation (d = 1, 45°)	1	1	1	1	1	1	1	1
286	LBP Statistical	Autocorrelation (d = 1, 45°)	1	0	1	0	0	0	0	0
287	LBP Statistical	Entropy (d = 1, 45°)	1	0	1	0	0	0	0	0
288	LBP Statistical	Measure of Correlation 1 (d = 1, 45°)	0	0	0	0	0	0	0	0
289	LBP Statistical	Measure of Correlation 2 (d = 1, 45°)	1	1	1	1	1	1	1	1
290	LBP Statistical	Cluster Shade (d = 1, 45°)	1	1	1	1	1	1	1	1
291	LBP Statistical	Cluster Prominence (d = 1, 45°)	1	1	1	1	1	1	1	1
292	LBP Statistical	Max Probability (d = 1, 45°)	1	0	1	0	0	0	0	0
293	LBP Statistical	Sum Average (d = 1, 45°)	1	1	1	1	1	1	1	1
294	LBP Statistical	Sum Entropy (d = 1, 45°)	1	0	1	0	1	1	1	0
295	LBP Statistical	Sum Variance (d = 1, 45°)	1	0	1	0	1	1	1	0
296	LBP Statistical	Difference Entropy (d = 1, 45°)	0	0	0	0	0	0	0	0
297	LBP Statistical	Difference Variance (d = 1, 45°)	1	0	0	0	0	0	0	0
298	LBP Statistical	Homogeneity 1 (d = 1, 45°)	0	0	0	0	0	0	0	0
299	LBP Statistical	Homogeneity 2 (d = 1, 45°)	0	0	0	0	0	0	0	0

300	LBP Statistical	Energy (d = 1, 90°)	1	0	1	0	0	0	0	0
301	LBP Statistical	Variance (d = 1, 90°)	1	1	1	1	1	1	1	1
302	LBP Statistical	Contrast (d = 1, 90°)	1	1	0	0	0	0	0	0
303	LBP Statistical	Dissimilarity (d = 1, 90°)	1	0	0	0	0	0	0	0
304	LBP Statistical	Correlation (d = 1, 90°)	1	1	1	1	1	1	1	1
305	LBP Statistical	Autocorrelation (d = 1, 90°)	1	0	1	0	0	0	0	0
306	LBP Statistical	Entropy (d = 1, 90°)	1	0	1	0	0	0	0	0
307	LBP Statistical	Measure of Correlation 1 (d = 1, 90°)	0	0	0	0	0	0	0	0
308	LBP Statistical	Measure of Correlation 2 (d = 1, 90°)	1	1	1	1	1	1	1	1
309	LBP Statistical	Cluster Shade (d = 1, 90°)	1	1	1	1	1	1	1	1
310	LBP Statistical	Cluster Prominence (d = 1, 90°)	1	1	1	1	1	1	1	1
311	LBP Statistical	Max Probability (d = 1, 90°)	1	0	1	0	0	0	0	0
312	LBP Statistical	Sum Average (d = 1, 90°)	1	1	1	1	1	1	1	1
313	LBP Statistical	Sum Entropy (d = 1, 90°)	1	0	1	0	1	0	0	0
314	LBP Statistical	Sum Variance (d = 1, 90°)	1	0	1	0	1	0	0	0
315	LBP Statistical	Difference Entropy (d = 1, 90°)	0	0	0	0	0	0	0	0
316	LBP Statistical	Difference Variance (d = 1, 90°)	1	0	0	0	0	0	0	0
317	LBP Statistical	Homogeneity 1 (d = 1, 90°)	0	0	0	0	0	0	0	0
318	LBP Statistical	Homogeneity 2 (d = 1, 90°)	0	0	0	0	0	0	0	0
319	LBP Statistical	Energy (d = 1, 135°)	1	0	1	0	0	0	0	0
320	LBP Statistical	Variance (d = 1, 135°)	1	1	1	1	1	1	1	1
321	LBP Statistical	Contrast (d = 1, 135°)	1	1	0	0	0	0	0	0
322	LBP Statistical	Dissimilarity (d = 1, 135°)	1	0	0	0	0	0	0	0
323	LBP Statistical	Correlation (d = 1, 135°)	1	1	1	1	1	1	1	1
324	LBP Statistical	Autocorrelation (d = 1, 135°)	1	0	1	0	0	0	0	0
325	LBP Statistical	Entropy (d = 1, 135°)	1	0	1	0	0	0	0	0
326	LBP Statistical	Measure of Correlation 1 (d = 1, 135°)	0	0	0	0	0	0	0	0
327	LBP Statistical	Measure of Correlation 2 (d = 1, 135°)	1	1	1	1	1	1	1	1
328	LBP Statistical	Cluster Shade (d = 1, 135°)	1	1	1	1	1	1	1	1
329	LBP Statistical	Cluster Prominence (d = 1, 135°)	1	1	1	1	1	1	1	1

330	LBP Statistical	Max Probability (d = 1, 135°)	1	0	1	0	0	0	0	0
331	LBP Statistical	Sum Average (d = 1, 135°)	1	1	1	1	1	1	1	1
332	LBP Statistical	Sum Entropy (d = 1, 135°)	1	0	1	1	1	1	1	0
333	LBP Statistical	Sum Variance (d = 1, 135°)	1	0	1	1	1	1	1	0
334	LBP Statistical	Difference Entropy (d = 1, 135°)	0	0	0	0	0	0	0	0
335	LBP Statistical	Difference Variance (d = 1, 135°)	1	0	0	0	0	0	0	0
336	LBP Statistical	Homogeneity 1 (d = 1, 135°)	0	0	0	0	0	0	0	0
337	LBP Statistical	Homogeneity 2 (d = 1, 135°)	0	0	0	0	0	0	0	0
338	LBP Statistical	Energy (d = 3, 0°)	1	0	1	0	1	0	0	0
339	LBP Statistical	Variance (d = 3, 0°)	1	1	1	1	1	1	1	1
340	LBP Statistical	Contrast (d = 3, 0°)	1	0	0	0	0	0	0	0
341	LBP Statistical	Dissimilarity (d = 3, 0°)	0	0	0	0	0	0	0	0
342	LBP Statistical	Correlation (d = 3, 0°)	1	1	1	1	1	1	1	1
343	LBP Statistical	Autocorrelation (d = 3, 0°)	1	0	1	0	0	0	0	0
344	LBP Statistical	Entropy (d = 3, 0°)	1	0	1	1	1	1	1	0
345	LBP Statistical	Measure of Correlation 1 (d = 3, 0°)	1	0	1	0	0	0	0	0
346	LBP Statistical	Measure of Correlation 2 (d = 3, 0°)	1	1	1	1	1	1	1	1
347	LBP Statistical	Cluster Shade (d = 3, 0°)	1	1	1	1	1	1	1	1
348	LBP Statistical	Cluster Prominence (d = 3, 0°)	1	1	1	1	1	1	1	1
349	LBP Statistical	Max Probability (d = 3, 0°)	1	0	1	0	0	0	0	0
350	LBP Statistical	Sum Average (d = 3, 0°)	1	1	1	1	1	1	1	1
351	LBP Statistical	Sum Entropy (d = 3, 0°)	1	1	1	1	1	1	1	1
352	LBP Statistical	Sum Variance (d = 3, 0°)	1	1	1	1	1	1	1	1
353	LBP Statistical	Difference Entropy (d = 3, 0°)	0	0	0	0	0	0	0	0
354	LBP Statistical	Difference Variance (d = 3, 0°)	1	0	0	0	0	0	0	0
355	LBP Statistical	Homogeneity 1 (d = 3, 0°)	0	0	0	0	0	0	0	0
356	LBP Statistical	Homogeneity 2 (d = 3, 0°)	0	0	0	0	0	0	0	0
357	LBP Statistical	Energy (d = 3, 45°)	1	0	1	0	1	0	0	0
358	LBP Statistical	Variance (d = 3, 45°)	1	1	1	1	1	1	1	1
359	LBP Statistical	Contrast (d = 3, 45°)	0	0	0	0	0	0	0	0

360	LBP Statistical	Dissimilarity (d = 3, 45°)	0	0	0	0	0	0	0	0
361	LBP Statistical	Correlation (d = 3, 45°)	1	1	0	0	0	0	0	0
362	LBP Statistical	Autocorrelation (d = 3, 45°)	1	0	1	0	0	0	0	0
363	LBP Statistical	Entropy (d = 3, 45°)	1	0	1	1	1	1	1	0
364	LBP Statistical	Measure of Correlation 1 (d = 3, 45°)	1	0	1	1	1	1	1	0
365	LBP Statistical	Measure of Correlation 2 (d = 3, 45°)	1	1	1	1	1	1	1	1
366	LBP Statistical	Cluster Shade (d = 3, 45°)	1	1	1	1	1	1	1	1
367	LBP Statistical	Cluster Prominence (d = 3, 45°)	1	1	1	1	1	1	1	1
368	LBP Statistical	Max Probability (d = 3, 45°)	1	0	1	0	0	0	0	0
369	LBP Statistical	Sum Average (d = 3, 45°)	1	1	1	1	1	1	1	1
370	LBP Statistical	Sum Entropy (d = 3, 45°)	1	1	1	1	1	1	1	1
371	LBP Statistical	Sum Variance (d = 3, 45°)	1	1	1	1	1	1	1	1
372	LBP Statistical	Difference Entropy (d = 3, 45°)	0	0	0	0	0	0	0	0
373	LBP Statistical	Difference Variance (d = 3, 45°)	0	0	0	0	0	0	0	0
374	LBP Statistical	Homogeneity 1 (d = 3, 45°)	0	0	0	0	0	0	0	0
375	LBP Statistical	Homogeneity 2 (d = 3, 45°)	0	0	0	0	0	0	0	0
376	LBP Statistical	Energy (d = 3, 90°)	1	0	1	0	0	0	0	0
377	LBP Statistical	Variance (d = 3, 90°)	1	1	1	1	1	1	1	1
378	LBP Statistical	Contrast (d = 3, 90°)	1	0	0	0	0	0	0	0
379	LBP Statistical	Dissimilarity (d = 3, 90°)	1	0	0	0	0	0	0	0
380	LBP Statistical	Correlation (d = 3, 90°)	1	1	1	1	1	1	1	1
381	LBP Statistical	Autocorrelation (d = 3, 90°)	1	0	1	0	0	0	0	0
382	LBP Statistical	Entropy (d = 3, 90°)	1	1	1	1	1	1	1	1
383	LBP Statistical	Measure of Correlation 1 (d = 3, 90°)	1	0	1	0	0	0	0	0
384	LBP Statistical	Measure of Correlation 2 (d = 3, 90°)	1	1	1	1	1	1	1	1
385	LBP Statistical	Cluster Shade (d = 3, 90°)	1	1	1	1	1	1	1	1
386	LBP Statistical	Cluster Prominence (d = 3, 90°)	1	1	1	1	1	1	1	1
387	LBP Statistical	Max Probability (d = 3, 90°)	1	0	1	0	0	0	0	0
388	LBP Statistical	Sum Average (d = 3, 90°)	1	1	1	1	1	1	1	1
389	LBP Statistical	Sum Entropy (d = 3, 90°)	1	1	1	1	1	1	1	1

390	LBP Statistical	Sum Variance (d = 3, 90°)	1	1	1	1	1	1	1	1
391	LBP Statistical	Difference Entropy (d = 3, 90°)	0	0	0	0	0	0	0	0
392	LBP Statistical	Difference Variance (d = 3, 90°)	1	0	0	0	0	0	0	0
393	LBP Statistical	Homogeneity 1 (d = 3, 90°)	0	0	0	0	0	0	0	0
394	LBP Statistical	Homogeneity 2 (d = 3, 90°)	0	0	0	0	0	0	0	0
395	LBP Statistical	Energy (d = 3, 135°)	1	1	1	0	1	0	0	0
396	LBP Statistical	Variance (d = 3, 135°)	1	1	1	1	1	1	1	1
397	LBP Statistical	Contrast (d = 3, 135°)	0	0	0	0	0	0	0	0
398	LBP Statistical	Dissimilarity (d = 3, 135°)	0	0	0	0	0	0	0	0
399	LBP Statistical	Correlation (d = 3, 135°)	1	1	0	0	0	0	0	0
400	LBP Statistical	Autocorrelation (d = 3, 135°)	1	1	1	0	0	0	0	0
401	LBP Statistical	Entropy (d = 3, 135°)	1	1	1	1	1	1	1	1
402	LBP Statistical	Measure of Correlation 1 (d = 3, 135°)	1	1	1	1	1	0	1	0
403	LBP Statistical	Measure of Correlation 2 (d = 3, 135°)	1	1	1	1	1	1	1	1
404	LBP Statistical	Cluster Shade (d = 3, 135°)	1	1	1	1	1	1	1	1
405	LBP Statistical	Cluster Prominence (d = 3, 135°)	1	1	1	1	1	1	1	1
406	LBP Statistical	Max Probability (d = 3, 135°)	1	1	1	0	0	0	0	0
407	LBP Statistical	Sum Average (d = 3, 135°)	1	1	1	1	1	1	1	1
408	LBP Statistical	Sum Entropy (d = 3, 135°)	1	1	1	1	1	1	1	1
409	LBP Statistical	Sum Variance (d = 3, 135°)	1	1	1	1	1	1	1	1
410	LBP Statistical	Difference Entropy (d = 3, 135°)	0	0	0	0	0	0	0	0
411	LBP Statistical	Difference Variance (d = 3, 135°)	0	0	0	0	0	0	0	0
412	LBP Statistical	Homogeneity 1 (d = 3, 135°)	0	0	0	0	0	0	0	0
413	LBP Statistical	Homogeneity 2 (d = 3, 135°)	0	0	0	0	0	0	0	0
414	LBP Statistical	Energy (d = 5, 0°)	1	0	1	0	1	0	0	0
415	LBP Statistical	Variance (d = 5, 0°)	1	1	1	1	1	1	1	1
416	LBP Statistical	Contrast (d = 5, 0°)	1	0	0	0	0	0	0	0
417	LBP Statistical	Dissimilarity (d = 5, 0°)	0	0	0	0	0	0	0	0
418	LBP Statistical	Correlation (d = 5, 0°)	1	1	1	1	1	1	1	1
419	LBP Statistical	Autocorrelation (d = 5, 0°)	1	0	1	0	0	0	0	0

420	LBP Statistical	Entropy (d = 5, 0°)	1	0	1	1	1	1	1	0
421	LBP Statistical	Measure of Correlation 1 (d = 5, 0°)	1	0	1	0	0	0	0	0
422	LBP Statistical	Measure of Correlation 2 (d = 5, 0°)	1	1	1	1	1	1	1	1
423	LBP Statistical	Cluster Shade (d = 5, 0°)	1	1	1	1	1	1	1	1
424	LBP Statistical	Cluster Prominence (d = 5, 0°)	1	1	1	1	1	1	1	1
425	LBP Statistical	Max Probability (d = 5, 0°)	1	0	1	0	0	0	0	0
426	LBP Statistical	Sum Average (d = 5, 0°)	1	1	1	1	1	1	1	1
427	LBP Statistical	Sum Entropy (d = 5, 0°)	1	1	1	1	1	1	1	1
428	LBP Statistical	Sum Variance (d = 5, 0°)	1	1	1	1	1	1	1	1
429	LBP Statistical	Difference Entropy (d = 5, 0°)	0	0	0	0	0	0	0	0
430	LBP Statistical	Difference Variance (d = 5, 0°)	0	0	0	0	0	0	0	0
431	LBP Statistical	Homogeneity 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
432	LBP Statistical	Homogeneity 2 (d = 5, 0°)	0	0	0	0	0	0	0	0
433	LBP Statistical	Energy (d = 5, 45°)	1	1	1	1	1	1	1	1
434	LBP Statistical	Variance (d = 5, 45°)	1	1	1	1	1	1	1	1
435	LBP Statistical	Contrast (d = 5, 45°)	0	0	0	0	0	0	0	0
436	LBP Statistical	Dissimilarity (d = 5, 45°)	0	0	0	0	0	0	0	0
437	LBP Statistical	Correlation (d = 5, 45°)	0	0	0	0	0	0	0	0
438	LBP Statistical	Autocorrelation (d = 5, 45°)	1	1	1	1	1	1	1	1
439	LBP Statistical	Entropy (d = 5, 45°)	1	0	1	1	1	1	1	0
440	LBP Statistical	Measure of Correlation 1 (d = 5, 45°)	1	0	1	0	1	1	1	0
441	LBP Statistical	Measure of Correlation 2 (d = 5, 45°)	1	1	1	1	1	1	1	1
442	LBP Statistical	Cluster Shade (d = 5, 45°)	1	1	1	1	1	1	1	1
443	LBP Statistical	Cluster Prominence (d = 5, 45°)	1	1	1	1	1	1	1	1
444	LBP Statistical	Max Probability (d = 5, 45°)	1	1	1	0	1	1	1	0
445	LBP Statistical	Sum Average (d = 5, 45°)	1	1	1	1	1	1	1	1
446	LBP Statistical	Sum Entropy (d = 5, 45°)	1	1	1	1	1	1	1	1
447	LBP Statistical	Sum Variance (d = 5, 45°)	1	1	1	1	1	1	1	1
448	LBP Statistical	Difference Entropy (d = 5, 45°)	1	0	0	0	0	0	0	0
449	LBP Statistical	Difference Variance (d = 5, 45°)	0	0	0	0	0	0	0	0

450	LBP Statistical	Homogeneity 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
451	LBP Statistical	Homogeneity 2 (d = 5, 45°)	0	0	0	0	0	0	0	0
452	LBP Statistical	Energy (d = 5, 90°)	1	1	1	0	1	0	1	0
453	LBP Statistical	Variance (d = 5, 90°)	1	1	1	1	1	1	1	1
454	LBP Statistical	Contrast (d = 5, 90°)	0	0	0	0	0	0	0	0
455	LBP Statistical	Dissimilarity (d = 5, 90°)	0	0	0	0	0	0	0	0
456	LBP Statistical	Correlation (d = 5, 90°)	1	0	0	0	0	0	0	0
457	LBP Statistical	Autocorrelation (d = 5, 90°)	1	1	1	0	1	0	0	0
458	LBP Statistical	Entropy (d = 5, 90°)	1	0	1	1	1	1	1	0
459	LBP Statistical	Measure of Correlation 1 (d = 5, 90°)	1	0	1	0	0	0	0	0
460	LBP Statistical	Measure of Correlation 2 (d = 5, 90°)	1	1	1	1	1	1	1	1
461	LBP Statistical	Cluster Shade (d = 5, 90°)	1	1	1	1	1	1	1	1
462	LBP Statistical	Cluster Prominence (d = 5, 90°)	1	1	1	1	1	1	1	1
463	LBP Statistical	Max Probability (d = 5, 90°)	1	1	1	0	1	0	0	0
464	LBP Statistical	Sum Average (d = 5, 90°)	1	1	1	1	1	1	1	1
465	LBP Statistical	Sum Entropy (d = 5, 90°)	1	1	1	1	1	1	1	1
466	LBP Statistical	Sum Variance (d = 5, 90°)	1	1	1	1	1	1	1	1
467	LBP Statistical	Difference Entropy (d = 5, 90°)	0	0	0	0	0	0	0	0
468	LBP Statistical	Difference Variance (d = 5, 90°)	0	0	0	0	0	0	0	0
469	LBP Statistical	Homogeneity 1 (d = 5, 90°)	0	0	0	0	0	0	0	0
470	LBP Statistical	Homogeneity 2 (d = 5, 90°)	0	0	0	0	0	0	0	0
471	LBP Statistical	Energy (d = 5, 135°)	1	1	1	0	1	1	1	0
472	LBP Statistical	Variance (d = 5, 135°)	1	1	1	1	1	1	1	1
473	LBP Statistical	Contrast (d = 5, 135°)	0	0	0	0	0	0	0	0
474	LBP Statistical	Dissimilarity (d = 5, 135°)	0	0	0	0	0	0	0	0
475	LBP Statistical	Correlation (d = 5, 135°)	0	0	0	0	0	0	0	0
476	LBP Statistical	Autocorrelation (d = 5, 135°)	1	1	1	0	1	0	0	0
477	LBP Statistical	Entropy (d = 5, 135°)	1	1	1	1	1	1	1	1
478	LBP Statistical	Measure of Correlation 1 (d = 5, 135°)	1	1	1	0	1	0	1	0
479	LBP Statistical	Measure of Correlation 2 (d = 5, 135°)	1	1	1	1	1	1	1	1

480	LBP Statistical	Cluster Shade (d = 5, 135°)	1	1	1	1	1	1	1	1
481	LBP Statistical	Cluster Prominence (d = 5, 135°)	1	1	1	1	1	1	1	1
482	LBP Statistical	Max Probability (d = 5, 135°)	1	1	1	0	1	1	1	0
483	LBP Statistical	Sum Average (d = 5, 135°)	1	1	1	1	1	1	1	1
484	LBP Statistical	Sum Entropy (d = 5, 135°)	1	1	1	1	1	1	1	1
485	LBP Statistical	Sum Variance (d = 5, 135°)	1	1	1	1	1	1	1	1
486	LBP Statistical	Difference Entropy (d = 5, 135°)	1	0	0	0	0	0	0	0
487	LBP Statistical	Difference Variance (d = 5, 135°)	0	0	0	0	0	0	0	0
488	LBP Statistical	Homogeneity 1 (d = 5, 135°)	0	0	0	0	0	0	0	0
489	LBP Statistical	Homogeneity 2 (d = 5, 135°)	0	0	0	0	0	0	0	0
490	Hu Moments	Hu Moment 1	0	0	0	0	0	0	0	0
491	Hu Moments	Hu Moment 2	1	1	1	1	1	1	1	1
492	Hu Moments	Hu Moment 3	0	0	0	0	0	0	0	0
493	Hu Moments	Hu Moment 4	0	1	0	0	0	0	0	0
494	Hu Moments	Hu Moment 5	0	0	0	0	0	0	0	0
495	Hu Moments	Hu Moment 6	0	0	0	0	0	0	0	0
496	Hu Moments	Hu Moment 7	0	0	0	0	0	0	0	0
497	LG1 Statistical	Mean	1	1	1	1	1	1	1	1
498	LG1 Statistical	Mode	1	1	1	1	1	1	1	1
499	LG1 Statistical	Min	1	1	1	1	1	1	1	1
500	LG1 Statistical	Max	1	0	0	1	0	0	1	0
501	LG1 Statistical	Variance	1	0	0	1	0	0	1	0
502	LG1 Statistical	Range	1	0	0	1	0	0	1	0
503	LG1 Statistical	Entropy	1	1	1	1	1	1	1	1
504	LG1 Statistical	1st Quartile	1	1	1	1	1	1	1	1
505	LG1 Statistical	2nd Quartile	1	1	1	1	1	1	1	1
506	LG1 Statistical	3rd Quartile	0	0	0	1	1	1	1	0
507	LG1 Statistical	Interquartile Range	0	0	0	1	1	1	1	0
508	LG1 Statistical	Asymmetry	1	1	1	1	1	1	1	1
509	LG1 Statistical	Kurtosis	1	0	0	0	0	0	0	0

510	LG1 Statistical	Energy (d = 1, 0°)	0	1	1	1	0	1	1	0
511	LG1 Statistical	Variance (d = 1, 0°)	1	1	1	1	1	1	1	1
512	LG1 Statistical	Contrast (d = 1, 0°)	0	1	1	0	0	0	0	0
513	LG1 Statistical	Dissimilarity (d = 1, 0°)	0	1	1	0	0	0	0	0
514	LG1 Statistical	Correlation (d = 1, 0°)	0	0	0	0	0	0	0	0
515	LG1 Statistical	Autocorrelation (d = 1, 0°)	1	1	1	1	1	1	1	1
516	LG1 Statistical	Entropy (d = 1, 0°)	0	1	1	1	0	1	1	0
517	LG1 Statistical	Measure of Correlation 1 (d = 1, 0°)	0	1	1	1	0	0	0	0
518	LG1 Statistical	Measure of Correlation 2 (d = 1, 0°)	1	1	1	1	1	1	1	1
519	LG1 Statistical	Cluster Shade (d = 1, 0°)	1	1	1	1	1	1	1	1
520	LG1 Statistical	Cluster Prominence (d = 1, 0°)	1	1	1	1	1	1	1	1
521	LG1 Statistical	Max Probability (d = 1, 0°)	0	1	1	1	0	1	1	0
522	LG1 Statistical	Sum Average (d = 1, 0°)	1	1	1	1	1	1	1	1
523	LG1 Statistical	Sum Entropy (d = 1, 0°)	0	1	1	1	0	1	1	0
524	LG1 Statistical	Sum Variance (d = 1, 0°)	0	1	1	1	0	1	1	0
525	LG1 Statistical	Difference Entropy (d = 1, 0°)	0	1	1	0	0	0	0	0
526	LG1 Statistical	Difference Variance (d = 1, 0°)	0	1	1	0	0	0	0	0
527	LG1 Statistical	Homogeneity 1 (d = 1, 0°)	0	1	1	0	0	0	0	0
528	LG1 Statistical	Homogeneity 2 (d = 1, 0°)	0	1	1	0	0	0	0	0
529	LG1 Statistical	Energy (d = 1, 45°)	0	1	1	1	0	1	1	0
530	LG1 Statistical	Variance (d = 1, 45°)	1	1	1	1	1	1	1	1
531	LG1 Statistical	Contrast (d = 1, 45°)	0	1	1	0	0	0	0	0
532	LG1 Statistical	Dissimilarity (d = 1, 45°)	0	1	1	0	0	0	0	0
533	LG1 Statistical	Correlation (d = 1, 45°)	0	0	0	0	0	0	0	0
534	LG1 Statistical	Autocorrelation (d = 1, 45°)	1	1	1	1	1	1	1	1
535	LG1 Statistical	Entropy (d = 1, 45°)	0	1	1	1	0	1	1	0
536	LG1 Statistical	Measure of Correlation 1 (d = 1, 45°)	0	1	1	1	0	1	0	0
537	LG1 Statistical	Measure of Correlation 2 (d = 1, 45°)	1	1	1	1	1	1	1	1
538	LG1 Statistical	Cluster Shade (d = 1, 45°)	1	1	1	1	1	1	1	1
539	LG1 Statistical	Cluster Prominence (d = 1, 45°)	1	1	1	1	1	1	1	1

540	LG1 Statistical	Max Probability (d = 1, 45°)	0	1	1	1	0	1	1	0
541	LG1 Statistical	Sum Average (d = 1, 45°)	1	1	1	1	1	1	1	1
542	LG1 Statistical	Sum Entropy (d = 1, 45°)	0	1	1	1	0	1	1	0
543	LG1 Statistical	Sum Variance (d = 1, 45°)	0	1	1	1	0	1	1	0
544	LG1 Statistical	Difference Entropy (d = 1, 45°)	0	1	1	0	0	0	0	0
545	LG1 Statistical	Difference Variance (d = 1, 45°)	0	1	1	0	0	0	0	0
546	LG1 Statistical	Homogeneity 1 (d = 1, 45°)	0	1	1	0	0	0	0	0
547	LG1 Statistical	Homogeneity 2 (d = 1, 45°)	0	1	1	0	0	0	0	0
548	LG1 Statistical	Energy (d = 1, 90°)	0	1	1	1	0	1	1	0
549	LG1 Statistical	Variance (d = 1, 90°)	1	1	1	1	1	1	1	1
550	LG1 Statistical	Contrast (d = 1, 90°)	0	1	1	0	0	0	0	0
551	LG1 Statistical	Dissimilarity (d = 1, 90°)	0	1	1	0	0	1	0	0
552	LG1 Statistical	Correlation (d = 1, 90°)	0	0	0	0	0	0	0	0
553	LG1 Statistical	Autocorrelation (d = 1, 90°)	1	1	1	1	1	1	1	1
554	LG1 Statistical	Entropy (d = 1, 90°)	0	1	1	1	0	1	1	0
555	LG1 Statistical	Measure of Correlation 1 (d = 1, 90°)	0	1	1	1	0	1	1	0
556	LG1 Statistical	Measure of Correlation 2 (d = 1, 90°)	1	1	1	1	1	1	1	1
557	LG1 Statistical	Cluster Shade (d = 1, 90°)	1	1	1	1	1	1	1	1
558	LG1 Statistical	Cluster Prominence (d = 1, 90°)	1	1	1	1	1	1	1	1
559	LG1 Statistical	Max Probability (d = 1, 90°)	0	1	1	1	0	1	1	0
560	LG1 Statistical	Sum Average (d = 1, 90°)	1	1	1	1	1	1	1	1
561	LG1 Statistical	Sum Entropy (d = 1, 90°)	0	1	1	1	0	1	1	0
562	LG1 Statistical	Sum Variance (d = 1, 90°)	0	1	1	1	0	1	1	0
563	LG1 Statistical	Difference Entropy (d = 1, 90°)	0	1	1	0	0	1	0	0
564	LG1 Statistical	Difference Variance (d = 1, 90°)	0	1	1	0	0	1	0	0
565	LG1 Statistical	Homogeneity 1 (d = 1, 90°)	0	1	1	0	0	1	0	0
566	LG1 Statistical	Homogeneity 2 (d = 1, 90°)	0	1	1	0	0	1	0	0
567	LG1 Statistical	Energy (d = 1, 135°)	0	1	1	1	0	1	1	0
568	LG1 Statistical	Variance (d = 1, 135°)	1	1	1	1	1	1	1	1
569	LG1 Statistical	Contrast (d = 1, 135°)	0	1	1	0	0	1	1	0

570	LG1 Statistical	Dissimilarity (d = 1, 135°)	0	1	1	0	0	1	1	0
571	LG1 Statistical	Correlation (d = 1, 135°)	0	0	0	0	0	0	0	0
572	LG1 Statistical	Autocorrelation (d = 1, 135°)	1	1	1	1	1	1	1	1
573	LG1 Statistical	Entropy (d = 1, 135°)	0	1	1	1	0	1	1	0
574	LG1 Statistical	Measure of Correlation 1 (d = 1, 135°)	0	1	1	1	0	1	1	0
575	LG1 Statistical	Measure of Correlation 2 (d = 1, 135°)	1	1	1	1	1	1	1	1
576	LG1 Statistical	Cluster Shade (d = 1, 135°)	1	1	1	1	1	1	1	1
577	LG1 Statistical	Cluster Prominence (d = 1, 135°)	1	1	1	1	1	1	1	1
578	LG1 Statistical	Max Probability (d = 1, 135°)	0	1	1	1	0	1	1	0
579	LG1 Statistical	Sum Average (d = 1, 135°)	1	1	1	1	1	1	1	1
580	LG1 Statistical	Sum Entropy (d = 1, 135°)	0	1	1	1	0	1	1	0
581	LG1 Statistical	Sum Variance (d = 1, 135°)	0	1	1	1	0	1	1	0
582	LG1 Statistical	Difference Entropy (d = 1, 135°)	0	1	1	0	0	1	1	0
583	LG1 Statistical	Difference Variance (d = 1, 135°)	0	1	1	0	0	1	1	0
584	LG1 Statistical	Homogeneity 1 (d = 1, 135°)	0	1	1	0	0	1	1	0
585	LG1 Statistical	Homogeneity 2 (d = 1, 135°)	0	1	1	0	0	1	1	0
586	LG1 Statistical	Energy (d = 3, 0°)	0	1	1	1	0	1	1	0
587	LG1 Statistical	Variance (d = 3, 0°)	1	1	1	1	1	1	1	1
588	LG1 Statistical	Contrast (d = 3, 0°)	0	1	1	0	0	1	0	0
589	LG1 Statistical	Dissimilarity (d = 3, 0°)	0	1	1	0	0	1	0	0
590	LG1 Statistical	Correlation (d = 3, 0°)	0	0	0	0	0	0	0	0
591	LG1 Statistical	Autocorrelation (d = 3, 0°)	1	1	1	1	1	1	1	1
592	LG1 Statistical	Entropy (d = 3, 0°)	0	1	1	1	0	1	1	0
593	LG1 Statistical	Measure of Correlation 1 (d = 3, 0°)	0	1	1	1	0	1	0	0
594	LG1 Statistical	Measure of Correlation 2 (d = 3, 0°)	1	1	1	1	1	1	1	1
595	LG1 Statistical	Cluster Shade (d = 3, 0°)	1	1	1	1	1	1	1	1
596	LG1 Statistical	Cluster Prominence (d = 3, 0°)	1	1	1	1	1	1	1	1
597	LG1 Statistical	Max Probability (d = 3, 0°)	0	1	1	1	0	1	1	0
598	LG1 Statistical	Sum Average (d = 3, 0°)	1	1	1	1	1	1	1	1
599	LG1 Statistical	Sum Entropy (d = 3, 0°)	0	1	1	1	0	1	1	0

600	LG1 Statistical	Sum Variance (d = 3, 0°)	0	1	1	1	0	1	1	0
601	LG1 Statistical	Difference Entropy (d = 3, 0°)	0	1	1	0	0	1	0	0
602	LG1 Statistical	Difference Variance (d = 3, 0°)	0	1	1	0	0	1	0	0
603	LG1 Statistical	Homogeneity 1 (d = 3, 0°)	0	1	1	0	0	1	0	0
604	LG1 Statistical	Homogeneity 2 (d = 3, 0°)	0	1	1	0	0	1	0	0
605	LG1 Statistical	Energy (d = 3, 45°)	0	1	1	1	0	1	1	0
606	LG1 Statistical	Variance (d = 3, 45°)	1	1	1	1	1	1	1	1
607	LG1 Statistical	Contrast (d = 3, 45°)	0	1	1	1	0	1	1	0
608	LG1 Statistical	Dissimilarity (d = 3, 45°)	0	1	1	1	0	1	0	0
609	LG1 Statistical	Correlation (d = 3, 45°)	0	0	0	0	0	0	0	0
610	LG1 Statistical	Autocorrelation (d = 3, 45°)	1	1	1	1	1	1	1	1
611	LG1 Statistical	Entropy (d = 3, 45°)	0	1	1	1	0	1	1	0
612	LG1 Statistical	Measure of Correlation 1 (d = 3, 45°)	0	1	1	1	0	1	1	0
613	LG1 Statistical	Measure of Correlation 2 (d = 3, 45°)	1	1	1	1	1	1	1	1
614	LG1 Statistical	Cluster Shade (d = 3, 45°)	1	1	1	1	1	1	1	1
615	LG1 Statistical	Cluster Prominence (d = 3, 45°)	1	1	1	1	1	1	1	1
616	LG1 Statistical	Max Probability (d = 3, 45°)	0	1	1	1	0	1	1	0
617	LG1 Statistical	Sum Average (d = 3, 45°)	1	1	1	1	1	1	1	1
618	LG1 Statistical	Sum Entropy (d = 3, 45°)	0	1	1	1	0	1	1	0
619	LG1 Statistical	Sum Variance (d = 3, 45°)	0	1	1	1	0	1	1	0
620	LG1 Statistical	Difference Entropy (d = 3, 45°)	0	1	1	1	0	1	0	0
621	LG1 Statistical	Difference Variance (d = 3, 45°)	0	1	1	1	0	1	0	0
622	LG1 Statistical	Homogeneity 1 (d = 3, 45°)	0	1	1	1	0	1	0	0
623	LG1 Statistical	Homogeneity 2 (d = 3, 45°)	0	1	1	1	0	1	0	0
624	LG1 Statistical	Energy (d = 3, 90°)	0	1	1	1	0	1	1	0
625	LG1 Statistical	Variance (d = 3, 90°)	1	1	1	1	1	1	1	1
626	LG1 Statistical	Contrast (d = 3, 90°)	0	1	1	1	0	1	1	0
627	LG1 Statistical	Dissimilarity (d = 3, 90°)	0	1	1	1	0	1	1	0
628	LG1 Statistical	Correlation (d = 3, 90°)	0	0	0	0	0	0	0	0
629	LG1 Statistical	Autocorrelation (d = 3, 90°)	1	1	1	1	1	1	1	1

630	LG1 Statistical	Entropy (d = 3, 90°)	0	1	1	1	0	1	1	0
631	LG1 Statistical	Measure of Correlation 1 (d = 3, 90°)	0	1	1	1	0	1	1	0
632	LG1 Statistical	Measure of Correlation 2 (d = 3, 90°)	1	1	1	1	1	1	1	1
633	LG1 Statistical	Cluster Shade (d = 3, 90°)	1	1	1	1	1	1	1	1
634	LG1 Statistical	Cluster Prominence (d = 3, 90°)	1	1	1	1	1	1	1	1
635	LG1 Statistical	Max Probability (d = 3, 90°)	0	1	1	1	0	1	1	0
636	LG1 Statistical	Sum Average (d = 3, 90°)	1	1	1	1	1	1	1	1
637	LG1 Statistical	Sum Entropy (d = 3, 90°)	0	1	1	1	0	1	1	0
638	LG1 Statistical	Sum Variance (d = 3, 90°)	0	1	1	1	0	1	1	0
639	LG1 Statistical	Difference Entropy (d = 3, 90°)	0	1	1	1	0	1	1	0
640	LG1 Statistical	Difference Variance (d = 3, 90°)	0	1	1	1	0	1	1	0
641	LG1 Statistical	Homogeneity 1 (d = 3, 90°)	0	1	1	1	0	1	1	0
642	LG1 Statistical	Homogeneity 2 (d = 3, 90°)	0	1	1	1	0	1	1	0
643	LG1 Statistical	Energy (d = 3, 135°)	0	1	1	1	0	1	1	0
644	LG1 Statistical	Variance (d = 3, 135°)	1	1	1	1	1	1	1	1
645	LG1 Statistical	Contrast (d = 3, 135°)	0	1	1	1	0	1	1	0
646	LG1 Statistical	Dissimilarity (d = 3, 135°)	0	1	1	1	0	1	1	0
647	LG1 Statistical	Correlation (d = 3, 135°)	0	0	0	0	0	0	0	0
648	LG1 Statistical	Autocorrelation (d = 3, 135°)	1	1	1	1	1	1	1	1
649	LG1 Statistical	Entropy (d = 3, 135°)	0	1	1	1	0	1	1	0
650	LG1 Statistical	Measure of Correlation 1 (d = 3, 135°)	0	1	1	1	0	1	1	0
651	LG1 Statistical	Measure of Correlation 2 (d = 3, 135°)	1	1	1	1	1	1	1	1
652	LG1 Statistical	Cluster Shade (d = 3, 135°)	1	1	1	1	1	1	1	1
653	LG1 Statistical	Cluster Prominence (d = 3, 135°)	1	1	1	1	1	1	1	1
654	LG1 Statistical	Max Probability (d = 3, 135°)	0	1	1	1	0	1	1	0
655	LG1 Statistical	Sum Average (d = 3, 135°)	1	1	1	1	1	1	1	1
656	LG1 Statistical	Sum Entropy (d = 3, 135°)	0	1	1	1	0	1	1	0
657	LG1 Statistical	Sum Variance (d = 3, 135°)	0	1	1	1	0	1	1	0
658	LG1 Statistical	Difference Entropy (d = 3, 135°)	0	1	1	1	0	1	1	0
659	LG1 Statistical	Difference Variance (d = 3, 135°)	0	1	1	1	0	1	1	0

660	LG1 Statistical	Homogeneity 1 (d = 3, 135°)	0	1	1	1	0	1	1	0
661	LG1 Statistical	Homogeneity 2 (d = 3, 135°)	0	1	1	1	0	1	1	0
662	LG1 Statistical	Energy (d = 5, 0°)	0	1	1	1	0	1	1	0
663	LG1 Statistical	Variance (d = 5, 0°)	1	1	1	1	1	1	1	1
664	LG1 Statistical	Contrast (d = 5, 0°)	0	1	1	1	0	1	1	0
665	LG1 Statistical	Dissimilarity (d = 5, 0°)	0	1	1	1	0	1	0	0
666	LG1 Statistical	Correlation (d = 5, 0°)	0	0	0	0	0	0	0	0
667	LG1 Statistical	Autocorrelation (d = 5, 0°)	1	1	1	1	1	1	1	1
668	LG1 Statistical	Entropy (d = 5, 0°)	0	1	1	1	0	1	1	0
669	LG1 Statistical	Measure of Correlation 1 (d = 5, 0°)	0	1	1	1	0	1	1	0
670	LG1 Statistical	Measure of Correlation 2 (d = 5, 0°)	1	1	1	1	1	1	1	1
671	LG1 Statistical	Cluster Shade (d = 5, 0°)	1	1	1	1	1	1	1	1
672	LG1 Statistical	Cluster Prominence (d = 5, 0°)	1	1	1	1	1	1	1	1
673	LG1 Statistical	Max Probability (d = 5, 0°)	0	1	1	1	0	1	1	0
674	LG1 Statistical	Sum Average (d = 5, 0°)	1	1	1	1	1	1	1	1
675	LG1 Statistical	Sum Entropy (d = 5, 0°)	0	1	1	1	0	1	1	0
676	LG1 Statistical	Sum Variance (d = 5, 0°)	0	1	1	1	0	1	1	0
677	LG1 Statistical	Difference Entropy (d = 5, 0°)	0	1	1	1	0	1	0	0
678	LG1 Statistical	Difference Variance (d = 5, 0°)	0	1	1	1	0	1	0	0
679	LG1 Statistical	Homogeneity 1 (d = 5, 0°)	0	1	1	1	0	1	0	0
680	LG1 Statistical	Homogeneity 2 (d = 5, 0°)	0	1	1	1	0	1	0	0
681	LG1 Statistical	Energy (d = 5, 45°)	0	1	1	1	0	1	1	0
682	LG1 Statistical	Variance (d = 5, 45°)	1	1	1	1	1	1	1	1
683	LG1 Statistical	Contrast (d = 5, 45°)	0	1	1	1	0	1	1	0
684	LG1 Statistical	Dissimilarity (d = 5, 45°)	0	1	1	1	0	1	1	0
685	LG1 Statistical	Correlation (d = 5, 45°)	0	1	0	0	0	0	0	0
686	LG1 Statistical	Autocorrelation (d = 5, 45°)	1	1	1	1	1	1	1	1
687	LG1 Statistical	Entropy (d = 5, 45°)	0	1	1	1	0	1	1	0
688	LG1 Statistical	Measure of Correlation 1 (d = 5, 45°)	0	1	1	1	0	1	1	0
689	LG1 Statistical	Measure of Correlation 2 (d = 5, 45°)	1	1	1	1	1	1	1	1

690	LG1 Statistical	Cluster Shade (d = 5, 45°)	1	1	1	1	1	1	1
691	LG1 Statistical	Cluster Prominence (d = 5, 45°)	1	1	1	1	1	1	1
692	LG1 Statistical	Max Probability (d = 5, 45°)	0	1	1	1	0	1	0
693	LG1 Statistical	Sum Average (d = 5, 45°)	1	1	1	1	1	1	1
694	LG1 Statistical	Sum Entropy (d = 5, 45°)	0	1	1	1	0	1	0
695	LG1 Statistical	Sum Variance (d = 5, 45°)	0	1	1	1	0	1	0
696	LG1 Statistical	Difference Entropy (d = 5, 45°)	0	1	1	1	0	1	0
697	LG1 Statistical	Difference Variance (d = 5, 45°)	0	1	1	1	0	1	0
698	LG1 Statistical	Homogeneity 1 (d = 5, 45°)	0	1	1	1	0	1	0
699	LG1 Statistical	Homogeneity 2 (d = 5, 45°)	0	1	1	1	0	1	0
700	LG1 Statistical	Energy (d = 5, 90°)	0	1	1	1	0	1	0
701	LG1 Statistical	Variance (d = 5, 90°)	1	1	1	1	1	1	1
702	LG1 Statistical	Contrast (d = 5, 90°)	0	1	1	1	0	1	0
703	LG1 Statistical	Dissimilarity (d = 5, 90°)	0	1	1	1	0	1	0
704	LG1 Statistical	Correlation (d = 5, 90°)	0	0	0	0	0	0	0
705	LG1 Statistical	Autocorrelation (d = 5, 90°)	1	1	1	1	1	1	1
706	LG1 Statistical	Entropy (d = 5, 90°)	0	1	1	1	0	1	0
707	LG1 Statistical	Measure of Correlation 1 (d = 5, 90°)	0	1	1	1	0	1	0
708	LG1 Statistical	Measure of Correlation 2 (d = 5, 90°)	1	1	1	1	1	1	1
709	LG1 Statistical	Cluster Shade (d = 5, 90°)	1	1	1	1	1	1	1
710	LG1 Statistical	Cluster Prominence (d = 5, 90°)	1	1	1	1	1	1	1
711	LG1 Statistical	Max Probability (d = 5, 90°)	0	1	1	1	0	1	0
712	LG1 Statistical	Sum Average (d = 5, 90°)	1	1	1	1	1	1	1
713	LG1 Statistical	Sum Entropy (d = 5, 90°)	0	1	1	1	0	1	0
714	LG1 Statistical	Sum Variance (d = 5, 90°)	0	1	1	1	0	1	0
715	LG1 Statistical	Difference Entropy (d = 5, 90°)	0	1	1	1	0	1	0
716	LG1 Statistical	Difference Variance (d = 5, 90°)	0	1	1	1	0	1	0
717	LG1 Statistical	Homogeneity 1 (d = 5, 90°)	0	1	1	1	0	1	0
718	LG1 Statistical	Homogeneity 2 (d = 5, 90°)	0	1	1	1	0	1	0
719	LG1 Statistical	Energy (d = 5, 135°)	0	1	1	1	0	1	0

720	LG1 Statistical	Variance (d = 5, 135°)	1	1	1	1	1	1	1
721	LG1 Statistical	Contrast (d = 5, 135°)	0	1	1	1	0	1	1
722	LG1 Statistical	Dissimilarity (d = 5, 135°)	0	1	1	1	0	1	1
723	LG1 Statistical	Correlation (d = 5, 135°)	0	0	0	0	0	0	0
724	LG1 Statistical	Autocorrelation (d = 5, 135°)	1	1	1	1	1	1	1
725	LG1 Statistical	Entropy (d = 5, 135°)	0	1	1	1	0	1	1
726	LG1 Statistical	Measure of Correlation 1 (d = 5, 135°)	0	1	1	1	0	1	1
727	LG1 Statistical	Measure of Correlation 2 (d = 5, 135°)	1	1	1	1	1	1	1
728	LG1 Statistical	Cluster Shade (d = 5, 135°)	1	1	1	1	1	1	1
729	LG1 Statistical	Cluster Prominence (d = 5, 135°)	1	1	1	1	1	1	1
730	LG1 Statistical	Max Probability (d = 5, 135°)	0	1	1	1	0	1	1
731	LG1 Statistical	Sum Average (d = 5, 135°)	1	1	1	1	1	1	1
732	LG1 Statistical	Sum Entropy (d = 5, 135°)	0	1	1	1	0	1	1
733	LG1 Statistical	Sum Variance (d = 5, 135°)	0	1	1	1	0	1	1
734	LG1 Statistical	Difference Entropy (d = 5, 135°)	0	1	1	1	0	1	1
735	LG1 Statistical	Difference Variance (d = 5, 135°)	0	1	1	1	0	1	1
736	LG1 Statistical	Homogeneity 1 (d = 5, 135°)	0	1	1	1	0	1	1
737	LG1 Statistical	Homogeneity 2 (d = 5, 135°)	0	1	1	1	0	1	1
738	LG2 Statistical	Mean	1	1	1	1	1	1	1
739	LG2 Statistical	Mode	1	1	1	1	1	1	1
740	LG2 Statistical	Min	1	1	1	1	1	1	1
741	LG2 Statistical	Max	0	0	0	0	0	0	0
742	LG2 Statistical	Variance	0	0	0	0	0	0	0
743	LG2 Statistical	Range	0	0	0	0	0	0	0
744	LG2 Statistical	Entropy	0	0	0	0	0	0	0
745	LG2 Statistical	1st Quartile	1	1	1	1	1	1	1
746	LG2 Statistical	2nd Quartile	1	1	1	1	1	1	1
747	LG2 Statistical	3rd Quartile	0	0	0	1	1	1	1
748	LG2 Statistical	Interquartile Range	0	0	0	1	1	1	1
749	LG2 Statistical	Asymmetry	0	0	0	0	0	0	0

750	LG2 Statistical	Kurtosis	0	0	0	0	0	0	0	0
751	LG2 Statistical	Energy (d = 1, 0°)	1	1	0	1	1	1	1	0
752	LG2 Statistical	Variance (d = 1, 0°)	1	1	1	1	1	1	1	1
753	LG2 Statistical	Contrast (d = 1, 0°)	1	1	0	0	1	0	0	0
754	LG2 Statistical	Dissimilarity (d = 1, 0°)	1	1	0	0	1	0	0	0
755	LG2 Statistical	Correlation (d = 1, 0°)	0	0	0	0	0	0	0	0
756	LG2 Statistical	Autocorrelation (d = 1, 0°)	1	1	1	1	1	1	1	1
757	LG2 Statistical	Entropy (d = 1, 0°)	0	1	0	1	1	1	1	0
758	LG2 Statistical	Measure of Correlation 1 (d = 1, 0°)	0	1	0	0	1	0	0	0
759	LG2 Statistical	Measure of Correlation 2 (d = 1, 0°)	1	1	1	1	1	1	1	1
760	LG2 Statistical	Cluster Shade (d = 1, 0°)	1	1	1	1	1	1	1	1
761	LG2 Statistical	Cluster Prominence (d = 1, 0°)	1	1	1	1	1	1	1	1
762	LG2 Statistical	Max Probability (d = 1, 0°)	1	1	0	1	1	1	1	0
763	LG2 Statistical	Sum Average (d = 1, 0°)	1	1	1	1	1	1	1	1
764	LG2 Statistical	Sum Entropy (d = 1, 0°)	0	1	0	1	1	1	1	0
765	LG2 Statistical	Sum Variance (d = 1, 0°)	0	1	0	1	1	1	1	0
766	LG2 Statistical	Difference Entropy (d = 1, 0°)	1	1	0	0	1	0	0	0
767	LG2 Statistical	Difference Variance (d = 1, 0°)	1	1	0	0	1	0	0	0
768	LG2 Statistical	Homogeneity 1 (d = 1, 0°)	1	1	0	0	1	0	0	0
769	LG2 Statistical	Homogeneity 2 (d = 1, 0°)	1	1	0	0	1	0	0	0
770	LG2 Statistical	Energy (d = 1, 45°)	1	1	0	1	1	1	1	0
771	LG2 Statistical	Variance (d = 1, 45°)	1	1	1	1	1	1	1	1
772	LG2 Statistical	Contrast (d = 1, 45°)	0	1	0	0	0	0	0	0
773	LG2 Statistical	Dissimilarity (d = 1, 45°)	0	1	0	0	0	0	0	0
774	LG2 Statistical	Correlation (d = 1, 45°)	1	1	1	1	1	1	1	1
775	LG2 Statistical	Autocorrelation (d = 1, 45°)	1	1	1	1	1	1	1	1
776	LG2 Statistical	Entropy (d = 1, 45°)	0	1	0	1	1	1	1	0
777	LG2 Statistical	Measure of Correlation 1 (d = 1, 45°)	0	1	0	0	0	0	0	0
778	LG2 Statistical	Measure of Correlation 2 (d = 1, 45°)	1	1	1	1	1	1	1	1
779	LG2 Statistical	Cluster Shade (d = 1, 45°)	1	1	1	1	1	1	1	1

780	LG2 Statistical	Cluster Prominence (d = 1, 45°)	1	1	1	1	1	1	1	1
781	LG2 Statistical	Max Probability (d = 1, 45°)	1	1	0	1	1	1	1	0
782	LG2 Statistical	Sum Average (d = 1, 45°)	1	1	1	1	1	1	1	1
783	LG2 Statistical	Sum Entropy (d = 1, 45°)	0	1	0	1	1	1	1	0
784	LG2 Statistical	Sum Variance (d = 1, 45°)	0	1	0	1	1	1	1	0
785	LG2 Statistical	Difference Entropy (d = 1, 45°)	0	1	0	0	0	0	0	0
786	LG2 Statistical	Difference Variance (d = 1, 45°)	0	1	0	0	0	0	0	0
787	LG2 Statistical	Homogeneity 1 (d = 1, 45°)	0	1	0	0	0	0	0	0
788	LG2 Statistical	Homogeneity 2 (d = 1, 45°)	0	1	0	0	0	0	0	0
789	LG2 Statistical	Energy (d = 1, 90°)	1	1	0	1	1	1	1	0
790	LG2 Statistical	Variance (d = 1, 90°)	1	1	1	1	1	1	1	1
791	LG2 Statistical	Contrast (d = 1, 90°)	1	1	1	1	1	1	1	1
792	LG2 Statistical	Dissimilarity (d = 1, 90°)	1	1	1	1	1	1	1	1
793	LG2 Statistical	Correlation (d = 1, 90°)	0	0	0	0	0	0	0	0
794	LG2 Statistical	Autocorrelation (d = 1, 90°)	1	1	1	1	1	1	1	1
795	LG2 Statistical	Entropy (d = 1, 90°)	1	1	0	1	1	1	1	0
796	LG2 Statistical	Measure of Correlation 1 (d = 1, 90°)	1	1	1	1	1	1	1	1
797	LG2 Statistical	Measure of Correlation 2 (d = 1, 90°)	1	1	1	1	1	1	1	1
798	LG2 Statistical	Cluster Shade (d = 1, 90°)	1	1	1	1	1	1	1	1
799	LG2 Statistical	Cluster Prominence (d = 1, 90°)	1	1	1	1	1	1	1	1
800	LG2 Statistical	Max Probability (d = 1, 90°)	1	1	0	1	1	1	1	0
801	LG2 Statistical	Sum Average (d = 1, 90°)	1	1	1	1	1	1	1	1
802	LG2 Statistical	Sum Entropy (d = 1, 90°)	1	1	0	1	1	1	1	0
803	LG2 Statistical	Sum Variance (d = 1, 90°)	1	1	0	1	1	1	1	0
804	LG2 Statistical	Difference Entropy (d = 1, 90°)	1	1	1	1	1	1	1	1
805	LG2 Statistical	Difference Variance (d = 1, 90°)	1	1	1	1	1	1	1	1
806	LG2 Statistical	Homogeneity 1 (d = 1, 90°)	1	1	1	1	1	1	1	1
807	LG2 Statistical	Homogeneity 2 (d = 1, 90°)	1	1	1	1	1	1	1	1
808	LG2 Statistical	Energy (d = 1, 135°)	1	1	0	1	1	1	1	0
809	LG2 Statistical	Variance (d = 1, 135°)	1	1	1	1	1	1	1	1

810	LG2 Statistical	Contrast (d = 1, 135°)	1	1	0	0	0	0	0	0
811	LG2 Statistical	Dissimilarity (d = 1, 135°)	1	1	0	0	0	0	0	0
812	LG2 Statistical	Correlation (d = 1, 135°)	0	0	1	0	0	0	0	0
813	LG2 Statistical	Autocorrelation (d = 1, 135°)	1	1	1	1	1	1	1	1
814	LG2 Statistical	Entropy (d = 1, 135°)	0	1	0	1	1	1	1	0
815	LG2 Statistical	Measure of Correlation 1 (d = 1, 135°)	0	1	0	0	0	0	0	0
816	LG2 Statistical	Measure of Correlation 2 (d = 1, 135°)	1	1	1	1	1	1	1	1
817	LG2 Statistical	Cluster Shade (d = 1, 135°)	1	1	1	1	1	1	1	1
818	LG2 Statistical	Cluster Prominence (d = 1, 135°)	1	1	1	1	1	1	1	1
819	LG2 Statistical	Max Probability (d = 1, 135°)	1	1	0	1	1	1	1	0
820	LG2 Statistical	Sum Average (d = 1, 135°)	1	1	1	1	1	1	1	1
821	LG2 Statistical	Sum Entropy (d = 1, 135°)	0	1	0	1	1	1	1	0
822	LG2 Statistical	Sum Variance (d = 1, 135°)	0	1	0	1	1	1	1	0
823	LG2 Statistical	Difference Entropy (d = 1, 135°)	1	1	0	0	0	0	0	0
824	LG2 Statistical	Difference Variance (d = 1, 135°)	1	1	0	0	0	0	0	0
825	LG2 Statistical	Homogeneity 1 (d = 1, 135°)	1	1	0	0	0	0	0	0
826	LG2 Statistical	Homogeneity 2 (d = 1, 135°)	1	1	0	0	0	0	0	0
827	LG2 Statistical	Energy (d = 3, 0°)	1	1	0	1	1	1	1	0
828	LG2 Statistical	Variance (d = 3, 0°)	1	1	1	1	1	1	1	1
829	LG2 Statistical	Contrast (d = 3, 0°)	0	1	0	0	0	0	0	0
830	LG2 Statistical	Dissimilarity (d = 3, 0°)	0	1	0	0	0	0	0	0
831	LG2 Statistical	Correlation (d = 3, 0°)	1	1	1	1	1	1	1	1
832	LG2 Statistical	Autocorrelation (d = 3, 0°)	1	1	1	1	1	1	1	1
833	LG2 Statistical	Entropy (d = 3, 0°)	0	1	0	1	1	1	1	0
834	LG2 Statistical	Measure of Correlation 1 (d = 3, 0°)	0	1	0	0	0	0	0	0
835	LG2 Statistical	Measure of Correlation 2 (d = 3, 0°)	1	1	1	1	1	1	1	1
836	LG2 Statistical	Cluster Shade (d = 3, 0°)	1	1	1	1	1	1	1	1
837	LG2 Statistical	Cluster Prominence (d = 3, 0°)	1	1	1	1	1	1	1	1
838	LG2 Statistical	Max Probability (d = 3, 0°)	1	1	0	1	1	1	1	0
839	LG2 Statistical	Sum Average (d = 3, 0°)	1	1	1	1	1	1	1	1

840	LG2 Statistical	Sum Entropy (d = 3, 0°)	0	1	0	1	1	1	1	0
841	LG2 Statistical	Sum Variance (d = 3, 0°)	0	1	0	1	1	1	1	0
842	LG2 Statistical	Difference Entropy (d = 3, 0°)	0	1	0	0	0	0	0	0
843	LG2 Statistical	Difference Variance (d = 3, 0°)	0	1	0	0	0	0	0	0
844	LG2 Statistical	Homogeneity 1 (d = 3, 0°)	0	1	0	0	0	0	0	0
845	LG2 Statistical	Homogeneity 2 (d = 3, 0°)	0	1	0	0	0	0	0	0
846	LG2 Statistical	Energy (d = 3, 45°)	0	1	0	1	1	1	1	0
847	LG2 Statistical	Variance (d = 3, 45°)	1	1	1	1	1	1	1	1
848	LG2 Statistical	Contrast (d = 3, 45°)	0	0	0	0	0	0	0	0
849	LG2 Statistical	Dissimilarity (d = 3, 45°)	0	1	0	0	0	0	0	0
850	LG2 Statistical	Correlation (d = 3, 45°)	1	1	1	1	1	1	1	1
851	LG2 Statistical	Autocorrelation (d = 3, 45°)	1	1	1	1	1	1	1	1
852	LG2 Statistical	Entropy (d = 3, 45°)	0	1	0	1	1	1	1	0
853	LG2 Statistical	Measure of Correlation 1 (d = 3, 45°)	0	1	0	0	0	0	0	0
854	LG2 Statistical	Measure of Correlation 2 (d = 3, 45°)	1	1	1	1	1	1	1	1
855	LG2 Statistical	Cluster Shade (d = 3, 45°)	1	1	1	1	1	1	1	1
856	LG2 Statistical	Cluster Prominence (d = 3, 45°)	1	1	1	1	1	1	1	1
857	LG2 Statistical	Max Probability (d = 3, 45°)	0	1	0	1	1	1	1	0
858	LG2 Statistical	Sum Average (d = 3, 45°)	1	1	1	1	1	1	1	1
859	LG2 Statistical	Sum Entropy (d = 3, 45°)	0	1	0	1	1	1	1	0
860	LG2 Statistical	Sum Variance (d = 3, 45°)	0	1	0	1	1	1	1	0
861	LG2 Statistical	Difference Entropy (d = 3, 45°)	0	1	0	0	0	0	0	0
862	LG2 Statistical	Difference Variance (d = 3, 45°)	0	0	0	0	0	0	0	0
863	LG2 Statistical	Homogeneity 1 (d = 3, 45°)	0	1	0	0	0	0	0	0
864	LG2 Statistical	Homogeneity 2 (d = 3, 45°)	0	1	0	0	0	0	0	0
865	LG2 Statistical	Energy (d = 3, 90°)	1	1	0	1	1	1	1	0
866	LG2 Statistical	Variance (d = 3, 90°)	1	1	1	1	1	1	1	1
867	LG2 Statistical	Contrast (d = 3, 90°)	0	1	0	0	0	0	0	0
868	LG2 Statistical	Dissimilarity (d = 3, 90°)	0	1	0	0	0	0	0	0
869	LG2 Statistical	Correlation (d = 3, 90°)	1	0	1	0	1	0	0	0

870	LG2 Statistical	Autocorrelation (d = 3, 90°)	1	1	1	1	1	1	1	1
871	LG2 Statistical	Entropy (d = 3, 90°)	0	1	0	1	1	1	1	0
872	LG2 Statistical	Measure of Correlation 1 (d = 3, 90°)	0	1	0	0	0	0	0	0
873	LG2 Statistical	Measure of Correlation 2 (d = 3, 90°)	1	1	1	1	1	1	1	1
874	LG2 Statistical	Cluster Shade (d = 3, 90°)	1	1	1	1	1	1	1	1
875	LG2 Statistical	Cluster Prominence (d = 3, 90°)	1	1	1	1	1	1	1	1
876	LG2 Statistical	Max Probability (d = 3, 90°)	1	1	0	1	1	1	1	0
877	LG2 Statistical	Sum Average (d = 3, 90°)	1	1	1	1	1	1	1	1
878	LG2 Statistical	Sum Entropy (d = 3, 90°)	0	1	0	1	1	1	1	0
879	LG2 Statistical	Sum Variance (d = 3, 90°)	0	1	0	1	1	1	1	0
880	LG2 Statistical	Difference Entropy (d = 3, 90°)	0	1	0	0	0	0	0	0
881	LG2 Statistical	Difference Variance (d = 3, 90°)	0	1	0	0	0	0	0	0
882	LG2 Statistical	Homogeneity 1 (d = 3, 90°)	0	1	0	0	0	0	0	0
883	LG2 Statistical	Homogeneity 2 (d = 3, 90°)	0	1	0	0	0	0	0	0
884	LG2 Statistical	Energy (d = 3, 135°)	1	1	0	1	1	1	1	0
885	LG2 Statistical	Variance (d = 3, 135°)	1	1	1	1	1	1	1	1
886	LG2 Statistical	Contrast (d = 3, 135°)	0	1	0	0	0	0	0	0
887	LG2 Statistical	Dissimilarity (d = 3, 135°)	0	1	0	0	0	0	0	0
888	LG2 Statistical	Correlation (d = 3, 135°)	1	1	1	1	1	1	1	1
889	LG2 Statistical	Autocorrelation (d = 3, 135°)	1	1	1	1	1	1	1	1
890	LG2 Statistical	Entropy (d = 3, 135°)	0	1	0	1	1	1	1	0
891	LG2 Statistical	Measure of Correlation 1 (d = 3, 135°)	0	1	0	0	0	0	0	0
892	LG2 Statistical	Measure of Correlation 2 (d = 3, 135°)	1	1	1	1	1	1	1	1
893	LG2 Statistical	Cluster Shade (d = 3, 135°)	1	1	1	1	1	1	1	1
894	LG2 Statistical	Cluster Prominence (d = 3, 135°)	1	1	1	1	1	1	1	1
895	LG2 Statistical	Max Probability (d = 3, 135°)	1	1	0	1	1	1	1	0
896	LG2 Statistical	Sum Average (d = 3, 135°)	1	1	1	1	1	1	1	1
897	LG2 Statistical	Sum Entropy (d = 3, 135°)	0	1	0	1	1	1	1	0
898	LG2 Statistical	Sum Variance (d = 3, 135°)	0	1	0	1	1	1	1	0
899	LG2 Statistical	Difference Entropy (d = 3, 135°)	0	1	0	0	0	0	0	0

900	LG2 Statistical	Difference Variance (d = 3, 135°)	0	1	0	0	0	0	0	0
901	LG2 Statistical	Homogeneity 1 (d = 3, 135°)	0	1	0	0	0	0	0	0
902	LG2 Statistical	Homogeneity 2 (d = 3, 135°)	0	1	0	0	0	0	0	0
903	LG2 Statistical	Energy (d = 5, 0°)	0	1	0	1	1	1	1	0
904	LG2 Statistical	Variance (d = 5, 0°)	1	1	1	1	1	1	1	1
905	LG2 Statistical	Contrast (d = 5, 0°)	0	1	0	0	0	0	0	0
906	LG2 Statistical	Dissimilarity (d = 5, 0°)	0	1	0	0	0	0	0	0
907	LG2 Statistical	Correlation (d = 5, 0°)	1	1	1	1	1	1	1	1
908	LG2 Statistical	Autocorrelation (d = 5, 0°)	1	1	1	1	1	1	1	1
909	LG2 Statistical	Entropy (d = 5, 0°)	0	1	0	1	1	1	1	0
910	LG2 Statistical	Measure of Correlation 1 (d = 5, 0°)	0	1	0	0	0	0	0	0
911	LG2 Statistical	Measure of Correlation 2 (d = 5, 0°)	1	1	1	1	1	1	1	1
912	LG2 Statistical	Cluster Shade (d = 5, 0°)	1	1	1	1	1	1	1	1
913	LG2 Statistical	Cluster Prominence (d = 5, 0°)	1	1	1	1	1	1	1	1
914	LG2 Statistical	Max Probability (d = 5, 0°)	0	1	0	1	1	1	1	0
915	LG2 Statistical	Sum Average (d = 5, 0°)	1	1	1	1	1	1	1	1
916	LG2 Statistical	Sum Entropy (d = 5, 0°)	0	1	0	1	1	1	1	0
917	LG2 Statistical	Sum Variance (d = 5, 0°)	0	1	0	1	1	1	1	0
918	LG2 Statistical	Difference Entropy (d = 5, 0°)	0	1	0	0	0	0	0	0
919	LG2 Statistical	Difference Variance (d = 5, 0°)	0	1	0	0	0	0	0	0
920	LG2 Statistical	Homogeneity 1 (d = 5, 0°)	0	1	0	0	0	0	0	0
921	LG2 Statistical	Homogeneity 2 (d = 5, 0°)	0	1	0	0	0	0	0	0
922	LG2 Statistical	Energy (d = 5, 45°)	0	1	0	1	1	1	1	0
923	LG2 Statistical	Variance (d = 5, 45°)	1	1	1	1	1	1	1	1
924	LG2 Statistical	Contrast (d = 5, 45°)	0	0	0	0	0	0	0	0
925	LG2 Statistical	Dissimilarity (d = 5, 45°)	0	0	0	0	0	0	0	0
926	LG2 Statistical	Correlation (d = 5, 45°)	1	1	1	1	1	1	1	1
927	LG2 Statistical	Autocorrelation (d = 5, 45°)	1	1	1	1	1	1	1	1
928	LG2 Statistical	Entropy (d = 5, 45°)	0	1	0	1	1	0	1	0
929	LG2 Statistical	Measure of Correlation 1 (d = 5, 45°)	0	0	0	0	0	0	0	0

930	LG2 Statistical	Measure of Correlation 2 (d = 5, 45°)	1	1	1	1	1	1	1	1
931	LG2 Statistical	Cluster Shade (d = 5, 45°)	1	1	1	1	1	1	1	1
932	LG2 Statistical	Cluster Prominence (d = 5, 45°)	1	1	1	1	1	1	1	1
933	LG2 Statistical	Max Probability (d = 5, 45°)	0	1	0	1	1	1	1	0
934	LG2 Statistical	Sum Average (d = 5, 45°)	1	1	1	1	1	1	1	1
935	LG2 Statistical	Sum Entropy (d = 5, 45°)	0	1	0	1	1	1	1	0
936	LG2 Statistical	Sum Variance (d = 5, 45°)	0	1	0	1	1	1	1	0
937	LG2 Statistical	Difference Entropy (d = 5, 45°)	0	0	0	0	0	0	0	0
938	LG2 Statistical	Difference Variance (d = 5, 45°)	0	0	0	0	0	0	0	0
939	LG2 Statistical	Homogeneity 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
940	LG2 Statistical	Homogeneity 2 (d = 5, 45°)	0	0	0	0	0	0	0	0
941	LG2 Statistical	Energy (d = 5, 90°)	1	1	0	1	1	1	1	0
942	LG2 Statistical	Variance (d = 5, 90°)	1	1	1	1	1	1	1	1
943	LG2 Statistical	Contrast (d = 5, 90°)	0	1	0	0	0	0	0	0
944	LG2 Statistical	Dissimilarity (d = 5, 90°)	0	1	0	0	0	0	0	0
945	LG2 Statistical	Correlation (d = 5, 90°)	1	1	1	1	1	1	1	1
946	LG2 Statistical	Autocorrelation (d = 5, 90°)	1	1	1	1	1	1	1	1
947	LG2 Statistical	Entropy (d = 5, 90°)	0	1	0	1	1	1	1	0
948	LG2 Statistical	Measure of Correlation 1 (d = 5, 90°)	0	1	0	0	0	0	0	0
949	LG2 Statistical	Measure of Correlation 2 (d = 5, 90°)	1	1	1	1	1	1	1	1
950	LG2 Statistical	Cluster Shade (d = 5, 90°)	1	1	1	1	1	1	1	1
951	LG2 Statistical	Cluster Prominence (d = 5, 90°)	1	1	1	1	1	1	1	1
952	LG2 Statistical	Max Probability (d = 5, 90°)	1	1	0	1	1	1	1	0
953	LG2 Statistical	Sum Average (d = 5, 90°)	1	1	1	1	1	1	1	1
954	LG2 Statistical	Sum Entropy (d = 5, 90°)	0	1	0	1	1	1	1	0
955	LG2 Statistical	Sum Variance (d = 5, 90°)	0	1	0	1	1	1	1	0
956	LG2 Statistical	Difference Entropy (d = 5, 90°)	0	1	0	0	0	0	0	0
957	LG2 Statistical	Difference Variance (d = 5, 90°)	0	1	0	0	0	0	0	0
958	LG2 Statistical	Homogeneity 1 (d = 5, 90°)	0	1	0	0	0	0	0	0
959	LG2 Statistical	Homogeneity 2 (d = 5, 90°)	0	1	0	0	0	0	0	0

960	LG2 Statistical	Energy (d = 5, 135°)	1	1	0	1	1	1	1	0
961	LG2 Statistical	Variance (d = 5, 135°)	1	1	1	1	1	1	1	1
962	LG2 Statistical	Contrast (d = 5, 135°)	0	1	0	0	0	0	0	0
963	LG2 Statistical	Dissimilarity (d = 5, 135°)	0	1	0	0	0	0	0	0
964	LG2 Statistical	Correlation (d = 5, 135°)	1	1	1	1	1	1	1	1
965	LG2 Statistical	Autocorrelation (d = 5, 135°)	1	1	1	1	1	1	1	1
966	LG2 Statistical	Entropy (d = 5, 135°)	0	1	0	1	1	1	1	0
967	LG2 Statistical	Measure of Correlation 1 (d = 5, 135°)	0	1	0	0	0	0	0	0
968	LG2 Statistical	Measure of Correlation 2 (d = 5, 135°)	1	1	1	1	1	1	1	1
969	LG2 Statistical	Cluster Shade (d = 5, 135°)	1	1	1	1	1	1	1	1
970	LG2 Statistical	Cluster Prominence (d = 5, 135°)	1	1	1	1	1	1	1	1
971	LG2 Statistical	Max Probability (d = 5, 135°)	1	1	0	1	1	1	1	0
972	LG2 Statistical	Sum Average (d = 5, 135°)	1	1	1	1	1	1	1	1
973	LG2 Statistical	Sum Entropy (d = 5, 135°)	0	1	0	1	1	1	1	0
974	LG2 Statistical	Sum Variance (d = 5, 135°)	0	1	0	1	1	1	1	0
975	LG2 Statistical	Difference Entropy (d = 5, 135°)	0	1	0	0	0	0	0	0
976	LG2 Statistical	Difference Variance (d = 5, 135°)	0	1	0	0	0	0	0	0
977	LG2 Statistical	Homogeneity 1 (d = 5, 135°)	0	1	0	0	0	0	0	0
978	LG2 Statistical	Homogeneity 2 (d = 5, 135°)	0	1	0	0	0	0	0	0
979	LG3 Statistical	Mean	1	1	1	1	1	1	1	1
980	LG3 Statistical	Mode	1	1	1	1	1	1	1	1
981	LG3 Statistical	Min	1	1	1	1	1	1	1	1
982	LG3 Statistical	Max	0	0	0	0	0	0	0	0
983	LG3 Statistical	Variance	0	0	0	0	0	0	0	0
984	LG3 Statistical	Range	0	0	0	0	0	0	0	0
985	LG3 Statistical	Entropy	0	0	0	0	0	0	0	0
986	LG3 Statistical	1st Quartile	1	1	1	1	1	1	1	1
987	LG3 Statistical	2nd Quartile	1	1	1	1	1	1	1	1
988	LG3 Statistical	3rd Quartile	0	0	0	0	0	0	0	0
989	LG3 Statistical	Interquartile Range	0	0	0	0	0	0	0	0

990	LG3 Statistical	Asymmetry	0	0	0	0	0	0	0	0
991	LG3 Statistical	Kurtosis	0	0	0	0	0	0	0	0
992	LG3 Statistical	Energy (d = 1, 0°)	1	1	0	0	1	1	1	0
993	LG3 Statistical	Variance (d = 1, 0°)	1	1	1	1	1	1	1	1
994	LG3 Statistical	Contrast (d = 1, 0°)	1	1	0	1	1	0	0	0
995	LG3 Statistical	Dissimilarity (d = 1, 0°)	1	1	0	1	1	0	0	0
996	LG3 Statistical	Correlation (d = 1, 0°)	0	0	0	0	0	0	0	0
997	LG3 Statistical	Autocorrelation (d = 1, 0°)	1	1	1	1	1	1	1	1
998	LG3 Statistical	Entropy (d = 1, 0°)	1	1	0	0	1	1	1	0
999	LG3 Statistical	Measure of Correlation 1 (d = 1, 0°)	1	1	0	0	1	0	0	0
1000	LG3 Statistical	Measure of Correlation 2 (d = 1, 0°)	1	1	1	1	1	1	1	1
1001	LG3 Statistical	Cluster Shade (d = 1, 0°)	1	1	1	1	1	1	1	1
1002	LG3 Statistical	Cluster Prominence (d = 1, 0°)	1	1	1	1	1	1	1	1
1003	LG3 Statistical	Max Probability (d = 1, 0°)	1	1	0	0	1	1	1	0
1004	LG3 Statistical	Sum Average (d = 1, 0°)	1	1	1	1	1	1	1	1
1005	LG3 Statistical	Sum Entropy (d = 1, 0°)	1	1	0	0	1	1	1	0
1006	LG3 Statistical	Sum Variance (d = 1, 0°)	1	1	0	0	1	1	1	0
1007	LG3 Statistical	Difference Entropy (d = 1, 0°)	1	1	0	1	1	0	0	0
1008	LG3 Statistical	Difference Variance (d = 1, 0°)	1	1	0	1	1	0	0	0
1009	LG3 Statistical	Homogeneity 1 (d = 1, 0°)	1	1	0	1	1	0	0	0
1010	LG3 Statistical	Homogeneity 2 (d = 1, 0°)	1	1	0	1	1	0	0	0
1011	LG3 Statistical	Energy (d = 1, 45°)	1	1	0	0	1	1	1	0
1012	LG3 Statistical	Variance (d = 1, 45°)	1	1	1	1	1	1	1	1
1013	LG3 Statistical	Contrast (d = 1, 45°)	1	1	0	0	0	0	0	0
1014	LG3 Statistical	Dissimilarity (d = 1, 45°)	1	1	0	0	0	0	0	0
1015	LG3 Statistical	Correlation (d = 1, 45°)	1	1	0	0	0	1	1	0
1016	LG3 Statistical	Autocorrelation (d = 1, 45°)	1	1	1	1	1	1	1	1
1017	LG3 Statistical	Entropy (d = 1, 45°)	1	1	0	0	1	0	1	0
1018	LG3 Statistical	Measure of Correlation 1 (d = 1, 45°)	1	1	0	0	0	0	0	0
1019	LG3 Statistical	Measure of Correlation 2 (d = 1, 45°)	1	1	1	1	1	1	1	1

1020	LG3 Statistical	Cluster Shade (d = 1, 45°)	1	1	1	1	1	1	1	1
1021	LG3 Statistical	Cluster Prominence (d = 1, 45°)	1	1	1	1	1	1	1	1
1022	LG3 Statistical	Max Probability (d = 1, 45°)	1	1	0	0	1	1	1	0
1023	LG3 Statistical	Sum Average (d = 1, 45°)	1	1	1	1	1	1	1	1
1024	LG3 Statistical	Sum Entropy (d = 1, 45°)	1	1	0	0	1	0	1	0
1025	LG3 Statistical	Sum Variance (d = 1, 45°)	1	1	0	0	1	0	1	0
1026	LG3 Statistical	Difference Entropy (d = 1, 45°)	1	1	0	0	0	0	0	0
1027	LG3 Statistical	Difference Variance (d = 1, 45°)	1	1	0	0	0	0	0	0
1028	LG3 Statistical	Homogeneity 1 (d = 1, 45°)	1	1	0	0	0	0	0	0
1029	LG3 Statistical	Homogeneity 2 (d = 1, 45°)	1	1	0	0	0	0	0	0
1030	LG3 Statistical	Energy (d = 1, 90°)	1	1	0	0	1	1	1	0
1031	LG3 Statistical	Variance (d = 1, 90°)	1	1	1	1	1	1	1	1
1032	LG3 Statistical	Contrast (d = 1, 90°)	1	1	0	1	1	1	1	0
1033	LG3 Statistical	Dissimilarity (d = 1, 90°)	1	1	0	1	1	1	1	0
1034	LG3 Statistical	Correlation (d = 1, 90°)	0	0	0	0	0	0	0	0
1035	LG3 Statistical	Autocorrelation (d = 1, 90°)	1	1	1	1	1	1	1	1
1036	LG3 Statistical	Entropy (d = 1, 90°)	1	1	0	0	1	1	1	0
1037	LG3 Statistical	Measure of Correlation 1 (d = 1, 90°)	1	1	0	1	1	1	1	0
1038	LG3 Statistical	Measure of Correlation 2 (d = 1, 90°)	1	1	1	1	1	1	1	1
1039	LG3 Statistical	Cluster Shade (d = 1, 90°)	1	1	1	1	1	1	1	1
1040	LG3 Statistical	Cluster Prominence (d = 1, 90°)	1	1	1	1	1	1	1	1
1041	LG3 Statistical	Max Probability (d = 1, 90°)	1	1	0	0	1	1	1	0
1042	LG3 Statistical	Sum Average (d = 1, 90°)	1	1	1	1	1	1	1	1
1043	LG3 Statistical	Sum Entropy (d = 1, 90°)	1	1	0	0	1	1	1	0
1044	LG3 Statistical	Sum Variance (d = 1, 90°)	1	1	0	0	1	1	1	0
1045	LG3 Statistical	Difference Entropy (d = 1, 90°)	1	1	0	1	1	1	1	0
1046	LG3 Statistical	Difference Variance (d = 1, 90°)	1	1	0	1	1	1	1	0
1047	LG3 Statistical	Homogeneity 1 (d = 1, 90°)	1	1	0	1	1	1	1	0
1048	LG3 Statistical	Homogeneity 2 (d = 1, 90°)	1	1	0	1	1	1	1	0
1049	LG3 Statistical	Energy (d = 1, 135°)	1	1	0	0	1	1	1	0

1050	LG3 Statistical	Variance (d = 1, 135°)	1	1	1	1	1	1	1
1051	LG3 Statistical	Contrast (d = 1, 135°)	1	1	0	0	0	0	0
1052	LG3 Statistical	Dissimilarity (d = 1, 135°)	1	1	0	0	0	0	0
1053	LG3 Statistical	Correlation (d = 1, 135°)	0	0	0	0	0	0	0
1054	LG3 Statistical	Autocorrelation (d = 1, 135°)	1	1	1	1	1	1	1
1055	LG3 Statistical	Entropy (d = 1, 135°)	1	1	0	0	1	1	0
1056	LG3 Statistical	Measure of Correlation 1 (d = 1, 135°)	1	1	0	0	0	0	0
1057	LG3 Statistical	Measure of Correlation 2 (d = 1, 135°)	1	1	1	1	1	1	1
1058	LG3 Statistical	Cluster Shade (d = 1, 135°)	1	1	1	1	1	1	1
1059	LG3 Statistical	Cluster Prominence (d = 1, 135°)	1	1	1	1	1	1	1
1060	LG3 Statistical	Max Probability (d = 1, 135°)	1	1	0	0	1	1	0
1061	LG3 Statistical	Sum Average (d = 1, 135°)	1	1	1	1	1	1	1
1062	LG3 Statistical	Sum Entropy (d = 1, 135°)	1	1	0	0	1	1	0
1063	LG3 Statistical	Sum Variance (d = 1, 135°)	1	1	0	0	1	1	0
1064	LG3 Statistical	Difference Entropy (d = 1, 135°)	1	1	0	0	0	0	0
1065	LG3 Statistical	Difference Variance (d = 1, 135°)	1	1	0	0	0	0	0
1066	LG3 Statistical	Homogeneity 1 (d = 1, 135°)	1	1	0	0	0	0	0
1067	LG3 Statistical	Homogeneity 2 (d = 1, 135°)	1	1	0	0	0	0	0
1068	LG3 Statistical	Energy (d = 3, 0°)	1	1	0	0	1	1	0
1069	LG3 Statistical	Variance (d = 3, 0°)	1	1	1	1	1	1	1
1070	LG3 Statistical	Contrast (d = 3, 0°)	0	1	0	0	0	0	0
1071	LG3 Statistical	Dissimilarity (d = 3, 0°)	0	1	0	0	0	0	0
1072	LG3 Statistical	Correlation (d = 3, 0°)	1	1	1	1	1	1	1
1073	LG3 Statistical	Autocorrelation (d = 3, 0°)	1	1	1	1	1	1	1
1074	LG3 Statistical	Entropy (d = 3, 0°)	1	1	0	0	0	0	0
1075	LG3 Statistical	Measure of Correlation 1 (d = 3, 0°)	0	1	0	0	0	0	0
1076	LG3 Statistical	Measure of Correlation 2 (d = 3, 0°)	1	1	1	1	1	1	1
1077	LG3 Statistical	Cluster Shade (d = 3, 0°)	1	1	1	1	1	1	1
1078	LG3 Statistical	Cluster Prominence (d = 3, 0°)	1	1	1	1	1	1	1
1079	LG3 Statistical	Max Probability (d = 3, 0°)	1	1	0	0	1	1	0

1080	LG3 Statistical	Sum Average (d = 3, 0°)	1	1	1	1	1	1	1
1081	LG3 Statistical	Sum Entropy (d = 3, 0°)	1	1	0	0	0	0	0
1082	LG3 Statistical	Sum Variance (d = 3, 0°)	1	1	0	0	0	0	0
1083	LG3 Statistical	Difference Entropy (d = 3, 0°)	0	1	0	0	0	0	0
1084	LG3 Statistical	Difference Variance (d = 3, 0°)	0	1	0	0	0	0	0
1085	LG3 Statistical	Homogeneity 1 (d = 3, 0°)	0	1	0	0	0	0	0
1086	LG3 Statistical	Homogeneity 2 (d = 3, 0°)	0	1	0	0	0	0	0
1087	LG3 Statistical	Energy (d = 3, 45°)	1	1	0	0	1	1	0
1088	LG3 Statistical	Variance (d = 3, 45°)	1	1	1	1	1	1	1
1089	LG3 Statistical	Contrast (d = 3, 45°)	0	0	0	0	0	0	0
1090	LG3 Statistical	Dissimilarity (d = 3, 45°)	0	0	0	0	0	0	0
1091	LG3 Statistical	Correlation (d = 3, 45°)	1	1	1	1	1	1	1
1092	LG3 Statistical	Autocorrelation (d = 3, 45°)	1	1	1	1	1	1	1
1093	LG3 Statistical	Entropy (d = 3, 45°)	1	1	0	0	0	0	0
1094	LG3 Statistical	Measure of Correlation 1 (d = 3, 45°)	0	1	0	0	0	0	0
1095	LG3 Statistical	Measure of Correlation 2 (d = 3, 45°)	1	1	1	1	1	1	1
1096	LG3 Statistical	Cluster Shade (d = 3, 45°)	1	1	1	1	1	1	1
1097	LG3 Statistical	Cluster Prominence (d = 3, 45°)	1	1	1	1	1	1	1
1098	LG3 Statistical	Max Probability (d = 3, 45°)	1	1	0	0	1	1	0
1099	LG3 Statistical	Sum Average (d = 3, 45°)	1	1	1	1	1	1	1
1100	LG3 Statistical	Sum Entropy (d = 3, 45°)	1	1	0	0	0	0	0
1101	LG3 Statistical	Sum Variance (d = 3, 45°)	1	1	0	0	0	0	0
1102	LG3 Statistical	Difference Entropy (d = 3, 45°)	0	0	0	0	0	0	0
1103	LG3 Statistical	Difference Variance (d = 3, 45°)	0	0	0	0	0	0	0
1104	LG3 Statistical	Homogeneity 1 (d = 3, 45°)	0	0	0	0	0	0	0
1105	LG3 Statistical	Homogeneity 2 (d = 3, 45°)	0	0	0	0	0	0	0
1106	LG3 Statistical	Energy (d = 3, 90°)	1	1	0	0	1	1	0
1107	LG3 Statistical	Variance (d = 3, 90°)	1	1	1	1	1	1	1
1108	LG3 Statistical	Contrast (d = 3, 90°)	1	1	0	0	1	0	0
1109	LG3 Statistical	Dissimilarity (d = 3, 90°)	1	1	0	0	1	0	0

1110	LG3 Statistical	Correlation (d = 3, 90°)	1	1	0	0	0	0	0	0
1111	LG3 Statistical	Autocorrelation (d = 3, 90°)	1	1	1	1	1	1	1	1
1112	LG3 Statistical	Entropy (d = 3, 90°)	1	1	0	0	1	1	1	0
1113	LG3 Statistical	Measure of Correlation 1 (d = 3, 90°)	1	1	0	0	0	0	0	0
1114	LG3 Statistical	Measure of Correlation 2 (d = 3, 90°)	1	1	1	1	1	1	1	1
1115	LG3 Statistical	Cluster Shade (d = 3, 90°)	1	1	1	1	1	1	1	1
1116	LG3 Statistical	Cluster Prominence (d = 3, 90°)	1	1	1	1	1	1	1	1
1117	LG3 Statistical	Max Probability (d = 3, 90°)	1	1	0	0	1	1	1	0
1118	LG3 Statistical	Sum Average (d = 3, 90°)	1	1	1	1	1	1	1	1
1119	LG3 Statistical	Sum Entropy (d = 3, 90°)	1	1	0	0	1	1	1	0
1120	LG3 Statistical	Sum Variance (d = 3, 90°)	1	1	0	0	1	1	1	0
1121	LG3 Statistical	Difference Entropy (d = 3, 90°)	1	1	0	0	1	0	0	0
1122	LG3 Statistical	Difference Variance (d = 3, 90°)	1	1	0	0	1	0	0	0
1123	LG3 Statistical	Homogeneity 1 (d = 3, 90°)	1	1	0	0	1	0	0	0
1124	LG3 Statistical	Homogeneity 2 (d = 3, 90°)	1	1	0	0	1	0	0	0
1125	LG3 Statistical	Energy (d = 3, 135°)	1	1	0	0	1	1	1	0
1126	LG3 Statistical	Variance (d = 3, 135°)	1	1	1	1	1	1	1	1
1127	LG3 Statistical	Contrast (d = 3, 135°)	1	0	0	0	0	0	0	0
1128	LG3 Statistical	Dissimilarity (d = 3, 135°)	1	0	0	0	0	0	0	0
1129	LG3 Statistical	Correlation (d = 3, 135°)	1	1	1	0	1	1	1	0
1130	LG3 Statistical	Autocorrelation (d = 3, 135°)	1	1	1	1	1	1	1	1
1131	LG3 Statistical	Entropy (d = 3, 135°)	1	1	0	0	1	0	1	0
1132	LG3 Statistical	Measure of Correlation 1 (d = 3, 135°)	0	0	0	0	0	0	0	0
1133	LG3 Statistical	Measure of Correlation 2 (d = 3, 135°)	1	1	1	1	1	1	1	1
1134	LG3 Statistical	Cluster Shade (d = 3, 135°)	1	1	1	1	1	1	1	1
1135	LG3 Statistical	Cluster Prominence (d = 3, 135°)	1	1	1	1	1	1	1	1
1136	LG3 Statistical	Max Probability (d = 3, 135°)	1	1	0	0	1	1	1	0
1137	LG3 Statistical	Sum Average (d = 3, 135°)	1	1	1	1	1	1	1	1
1138	LG3 Statistical	Sum Entropy (d = 3, 135°)	1	1	0	0	1	0	1	0
1139	LG3 Statistical	Sum Variance (d = 3, 135°)	1	1	0	0	1	0	1	0

1140	LG3 Statistical	Difference Entropy (d = 3, 135°)	1	0	0	0	0	0	0	0
1141	LG3 Statistical	Difference Variance (d = 3, 135°)	1	0	0	0	0	0	0	0
1142	LG3 Statistical	Homogeneity 1 (d = 3, 135°)	1	0	0	0	0	0	0	0
1143	LG3 Statistical	Homogeneity 2 (d = 3, 135°)	1	0	0	0	0	0	0	0
1144	LG3 Statistical	Energy (d = 5, 0°)	1	1	0	0	1	0	1	0
1145	LG3 Statistical	Variance (d = 5, 0°)	1	1	1	1	1	1	1	1
1146	LG3 Statistical	Contrast (d = 5, 0°)	0	0	0	0	0	0	0	0
1147	LG3 Statistical	Dissimilarity (d = 5, 0°)	0	0	0	0	0	0	0	0
1148	LG3 Statistical	Correlation (d = 5, 0°)	1	1	1	1	1	1	1	1
1149	LG3 Statistical	Autocorrelation (d = 5, 0°)	1	1	1	1	1	1	1	1
1150	LG3 Statistical	Entropy (d = 5, 0°)	1	1	0	0	0	0	0	0
1151	LG3 Statistical	Measure of Correlation 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
1152	LG3 Statistical	Measure of Correlation 2 (d = 5, 0°)	1	1	1	1	1	1	1	1
1153	LG3 Statistical	Cluster Shade (d = 5, 0°)	1	1	1	1	1	1	1	1
1154	LG3 Statistical	Cluster Prominence (d = 5, 0°)	1	1	1	1	1	1	1	1
1155	LG3 Statistical	Max Probability (d = 5, 0°)	1	1	0	0	1	1	1	0
1156	LG3 Statistical	Sum Average (d = 5, 0°)	1	1	1	1	1	1	1	1
1157	LG3 Statistical	Sum Entropy (d = 5, 0°)	1	1	0	0	0	0	0	0
1158	LG3 Statistical	Sum Variance (d = 5, 0°)	1	1	0	0	0	0	0	0
1159	LG3 Statistical	Difference Entropy (d = 5, 0°)	0	0	0	0	0	0	0	0
1160	LG3 Statistical	Difference Variance (d = 5, 0°)	0	0	0	0	0	0	0	0
1161	LG3 Statistical	Homogeneity 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
1162	LG3 Statistical	Homogeneity 2 (d = 5, 0°)	0	0	0	0	0	0	0	0
1163	LG3 Statistical	Energy (d = 5, 45°)	1	1	0	0	0	0	0	0
1164	LG3 Statistical	Variance (d = 5, 45°)	1	1	1	1	1	1	1	1
1165	LG3 Statistical	Contrast (d = 5, 45°)	0	0	0	0	0	0	0	0
1166	LG3 Statistical	Dissimilarity (d = 5, 45°)	0	0	0	0	0	0	0	0
1167	LG3 Statistical	Correlation (d = 5, 45°)	1	1	1	1	1	1	1	1
1168	LG3 Statistical	Autocorrelation (d = 5, 45°)	1	1	1	1	1	1	1	1
1169	LG3 Statistical	Entropy (d = 5, 45°)	1	1	0	0	0	0	0	0

1170	LG3 Statistical	Measure of Correlation 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
1171	LG3 Statistical	Measure of Correlation 2 (d = 5, 45°)	1	1	1	1	1	1	1	1
1172	LG3 Statistical	Cluster Shade (d = 5, 45°)	1	1	1	1	1	1	1	1
1173	LG3 Statistical	Cluster Prominence (d = 5, 45°)	1	1	1	1	1	1	1	1
1174	LG3 Statistical	Max Probability (d = 5, 45°)	1	1	0	0	0	0	0	0
1175	LG3 Statistical	Sum Average (d = 5, 45°)	1	1	1	1	1	1	1	1
1176	LG3 Statistical	Sum Entropy (d = 5, 45°)	1	1	0	0	0	0	0	0
1177	LG3 Statistical	Sum Variance (d = 5, 45°)	1	1	0	0	0	0	0	0
1178	LG3 Statistical	Difference Entropy (d = 5, 45°)	0	0	0	0	0	0	0	0
1179	LG3 Statistical	Difference Variance (d = 5, 45°)	0	0	0	0	0	0	0	0
1180	LG3 Statistical	Homogeneity 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
1181	LG3 Statistical	Homogeneity 2 (d = 5, 45°)	0	0	0	0	0	0	0	0
1182	LG3 Statistical	Energy (d = 5, 90°)	1	1	0	0	1	1	1	0
1183	LG3 Statistical	Variance (d = 5, 90°)	1	1	1	1	1	1	1	1
1184	LG3 Statistical	Contrast (d = 5, 90°)	0	1	0	0	0	0	0	0
1185	LG3 Statistical	Dissimilarity (d = 5, 90°)	0	1	0	0	0	0	0	0
1186	LG3 Statistical	Correlation (d = 5, 90°)	1	1	1	1	1	1	1	1
1187	LG3 Statistical	Autocorrelation (d = 5, 90°)	1	1	1	1	1	1	1	1
1188	LG3 Statistical	Entropy (d = 5, 90°)	1	1	0	0	1	0	0	0
1189	LG3 Statistical	Measure of Correlation 1 (d = 5, 90°)	1	1	0	0	0	0	0	0
1190	LG3 Statistical	Measure of Correlation 2 (d = 5, 90°)	1	1	1	1	1	1	1	1
1191	LG3 Statistical	Cluster Shade (d = 5, 90°)	1	1	1	1	1	1	1	1
1192	LG3 Statistical	Cluster Prominence (d = 5, 90°)	1	1	1	1	1	1	1	1
1193	LG3 Statistical	Max Probability (d = 5, 90°)	1	1	0	0	1	1	1	0
1194	LG3 Statistical	Sum Average (d = 5, 90°)	1	1	1	1	1	1	1	1
1195	LG3 Statistical	Sum Entropy (d = 5, 90°)	1	1	0	0	1	0	0	0
1196	LG3 Statistical	Sum Variance (d = 5, 90°)	1	1	0	0	1	0	0	0
1197	LG3 Statistical	Difference Entropy (d = 5, 90°)	0	1	0	0	0	0	0	0
1198	LG3 Statistical	Difference Variance (d = 5, 90°)	0	1	0	0	0	0	0	0
1199	LG3 Statistical	Homogeneity 1 (d = 5, 90°)	0	1	0	0	0	0	0	0

1200	LG3 Statistical	Homogeneity 2 (d = 5, 90°)	0	1	0	0	0	0	0	0
1201	LG3 Statistical	Energy (d = 5, 135°)	1	1	0	0	1	1	1	0
1202	LG3 Statistical	Variance (d = 5, 135°)	1	1	1	1	1	1	1	1
1203	LG3 Statistical	Contrast (d = 5, 135°)	0	0	0	0	0	0	0	0
1204	LG3 Statistical	Dissimilarity (d = 5, 135°)	0	0	0	0	0	0	0	0
1205	LG3 Statistical	Correlation (d = 5, 135°)	1	1	1	1	1	1	1	1
1206	LG3 Statistical	Autocorrelation (d = 5, 135°)	1	1	1	1	1	1	1	1
1207	LG3 Statistical	Entropy (d = 5, 135°)	1	1	0	0	0	0	0	0
1208	LG3 Statistical	Measure of Correlation 1 (d = 5, 135°)	0	0	0	0	0	0	0	0
1209	LG3 Statistical	Measure of Correlation 2 (d = 5, 135°)	1	1	1	1	1	1	1	1
1210	LG3 Statistical	Cluster Shade (d = 5, 135°)	1	1	1	1	1	1	1	1
1211	LG3 Statistical	Cluster Prominence (d = 5, 135°)	1	1	1	1	1	1	1	1
1212	LG3 Statistical	Max Probability (d = 5, 135°)	1	1	0	0	1	1	1	0
1213	LG3 Statistical	Sum Average (d = 5, 135°)	1	1	1	1	1	1	1	1
1214	LG3 Statistical	Sum Entropy (d = 5, 135°)	1	1	0	0	0	0	0	0
1215	LG3 Statistical	Sum Variance (d = 5, 135°)	1	1	0	0	0	0	0	0
1216	LG3 Statistical	Difference Entropy (d = 5, 135°)	0	0	0	0	0	0	0	0
1217	LG3 Statistical	Difference Variance (d = 5, 135°)	0	0	0	0	0	0	0	0
1218	LG3 Statistical	Homogeneity 1 (d = 5, 135°)	0	0	0	0	0	0	0	0
1219	LG3 Statistical	Homogeneity 2 (d = 5, 135°)	0	0	0	0	0	0	0	0
1220	LG4 Statistical	Mean	1	1	1	1	1	1	1	1
1221	LG4 Statistical	Mode	1	1	1	1	1	1	1	1
1222	LG4 Statistical	Min	1	1	1	1	1	1	1	1
1223	LG4 Statistical	Max	0	0	0	0	0	0	0	0
1224	LG4 Statistical	Variance	0	0	0	0	0	0	0	0
1225	LG4 Statistical	Range	0	0	0	0	0	0	0	0
1226	LG4 Statistical	Entropy	0	0	0	0	0	0	0	0
1227	LG4 Statistical	1st Quartile	1	1	1	1	1	1	1	1
1228	LG4 Statistical	2nd Quartile	1	1	1	1	1	1	1	1
1229	LG4 Statistical	3rd Quartile	0	0	0	0	0	0	0	0

1230	LG4 Statistical	Interquartile Range	0	0	0	0	0	0	0	0
1231	LG4 Statistical	Asymmetry	0	0	0	0	0	0	0	0
1232	LG4 Statistical	Kurtosis	0	0	0	0	0	0	0	0
1233	LG4 Statistical	Energy (d = 1, 0°)	1	0	0	0	1	0	0	0
1234	LG4 Statistical	Variance (d = 1, 0°)	1	1	1	1	1	1	1	1
1235	LG4 Statistical	Contrast (d = 1, 0°)	1	0	0	0	0	0	0	0
1236	LG4 Statistical	Dissimilarity (d = 1, 0°)	1	0	0	0	0	0	0	0
1237	LG4 Statistical	Correlation (d = 1, 0°)	1	0	0	0	0	0	0	0
1238	LG4 Statistical	Autocorrelation (d = 1, 0°)	1	1	1	1	1	1	1	1
1239	LG4 Statistical	Entropy (d = 1, 0°)	1	0	0	0	0	0	0	0
1240	LG4 Statistical	Measure of Correlation 1 (d = 1, 0°)	1	0	0	0	0	0	0	0
1241	LG4 Statistical	Measure of Correlation 2 (d = 1, 0°)	1	1	1	1	1	1	1	1
1242	LG4 Statistical	Cluster Shade (d = 1, 0°)	1	1	1	1	1	1	1	1
1243	LG4 Statistical	Cluster Prominence (d = 1, 0°)	1	1	1	1	1	1	1	1
1244	LG4 Statistical	Max Probability (d = 1, 0°)	1	0	0	0	1	0	0	0
1245	LG4 Statistical	Sum Average (d = 1, 0°)	1	1	1	1	1	1	1	1
1246	LG4 Statistical	Sum Entropy (d = 1, 0°)	1	0	0	0	0	0	0	0
1247	LG4 Statistical	Sum Variance (d = 1, 0°)	1	0	0	0	0	0	0	0
1248	LG4 Statistical	Difference Entropy (d = 1, 0°)	1	0	0	0	0	0	0	0
1249	LG4 Statistical	Difference Variance (d = 1, 0°)	1	0	0	0	0	0	0	0
1250	LG4 Statistical	Homogeneity 1 (d = 1, 0°)	1	0	0	0	0	0	0	0
1251	LG4 Statistical	Homogeneity 2 (d = 1, 0°)	1	0	0	0	0	0	0	0
1252	LG4 Statistical	Energy (d = 1, 45°)	1	0	0	0	1	0	0	0
1253	LG4 Statistical	Variance (d = 1, 45°)	1	1	1	1	1	1	1	1
1254	LG4 Statistical	Contrast (d = 1, 45°)	0	0	0	0	0	0	0	0
1255	LG4 Statistical	Dissimilarity (d = 1, 45°)	0	0	0	0	0	0	0	0
1256	LG4 Statistical	Correlation (d = 1, 45°)	1	1	1	0	0	0	0	0
1257	LG4 Statistical	Autocorrelation (d = 1, 45°)	1	1	1	1	1	1	1	1
1258	LG4 Statistical	Entropy (d = 1, 45°)	1	0	0	0	0	0	0	0
1259	LG4 Statistical	Measure of Correlation 1 (d = 1, 45°)	0	0	0	0	0	0	0	0

1260	LG4 Statistical	Measure of Correlation 2 (d = 1, 45°)	1	1	1	1	1	1	1
1261	LG4 Statistical	Cluster Shade (d = 1, 45°)	1	1	1	1	1	1	1
1262	LG4 Statistical	Cluster Prominence (d = 1, 45°)	1	1	1	1	1	1	1
1263	LG4 Statistical	Max Probability (d = 1, 45°)	1	0	0	0	1	0	0
1264	LG4 Statistical	Sum Average (d = 1, 45°)	1	1	1	1	1	1	1
1265	LG4 Statistical	Sum Entropy (d = 1, 45°)	1	0	0	0	0	0	0
1266	LG4 Statistical	Sum Variance (d = 1, 45°)	1	0	0	0	0	0	0
1267	LG4 Statistical	Difference Entropy (d = 1, 45°)	0	0	0	0	0	0	0
1268	LG4 Statistical	Difference Variance (d = 1, 45°)	0	0	0	0	0	0	0
1269	LG4 Statistical	Homogeneity 1 (d = 1, 45°)	0	0	0	0	0	0	0
1270	LG4 Statistical	Homogeneity 2 (d = 1, 45°)	0	0	0	0	0	0	0
1271	LG4 Statistical	Energy (d = 1, 90°)	1	0	0	0	1	0	0
1272	LG4 Statistical	Variance (d = 1, 90°)	1	1	1	1	1	1	1
1273	LG4 Statistical	Contrast (d = 1, 90°)	1	0	0	1	1	1	0
1274	LG4 Statistical	Dissimilarity (d = 1, 90°)	1	0	0	1	1	1	0
1275	LG4 Statistical	Correlation (d = 1, 90°)	0	0	0	0	0	0	0
1276	LG4 Statistical	Autocorrelation (d = 1, 90°)	1	1	1	1	1	1	1
1277	LG4 Statistical	Entropy (d = 1, 90°)	1	0	0	0	1	0	0
1278	LG4 Statistical	Measure of Correlation 1 (d = 1, 90°)	1	0	0	0	1	0	0
1279	LG4 Statistical	Measure of Correlation 2 (d = 1, 90°)	1	1	1	1	1	1	1
1280	LG4 Statistical	Cluster Shade (d = 1, 90°)	1	1	1	1	1	1	1
1281	LG4 Statistical	Cluster Prominence (d = 1, 90°)	1	1	1	1	1	1	1
1282	LG4 Statistical	Max Probability (d = 1, 90°)	1	0	0	0	1	0	0
1283	LG4 Statistical	Sum Average (d = 1, 90°)	1	1	1	1	1	1	1
1284	LG4 Statistical	Sum Entropy (d = 1, 90°)	1	0	0	0	1	0	0
1285	LG4 Statistical	Sum Variance (d = 1, 90°)	1	0	0	0	1	0	0
1286	LG4 Statistical	Difference Entropy (d = 1, 90°)	1	0	0	1	1	1	0
1287	LG4 Statistical	Difference Variance (d = 1, 90°)	1	0	0	1	1	1	0
1288	LG4 Statistical	Homogeneity 1 (d = 1, 90°)	1	0	0	1	1	1	0
1289	LG4 Statistical	Homogeneity 2 (d = 1, 90°)	1	0	0	1	1	1	0

1290	LG4 Statistical	Energy (d = 1, 135°)	1	0	0	0	1	0	0	0
1291	LG4 Statistical	Variance (d = 1, 135°)	1	1	1	1	1	1	1	1
1292	LG4 Statistical	Contrast (d = 1, 135°)	1	0	0	0	0	0	0	0
1293	LG4 Statistical	Dissimilarity (d = 1, 135°)	1	0	0	0	0	0	0	0
1294	LG4 Statistical	Correlation (d = 1, 135°)	1	1	1	0	0	0	0	0
1295	LG4 Statistical	Autocorrelation (d = 1, 135°)	1	1	1	1	1	1	1	1
1296	LG4 Statistical	Entropy (d = 1, 135°)	1	0	0	0	1	0	0	0
1297	LG4 Statistical	Measure of Correlation 1 (d = 1, 135°)	1	0	0	0	0	0	0	0
1298	LG4 Statistical	Measure of Correlation 2 (d = 1, 135°)	1	1	1	1	1	1	1	1
1299	LG4 Statistical	Cluster Shade (d = 1, 135°)	1	1	1	1	1	1	1	1
1300	LG4 Statistical	Cluster Prominence (d = 1, 135°)	1	1	1	1	1	1	1	1
1301	LG4 Statistical	Max Probability (d = 1, 135°)	1	0	0	0	1	0	0	0
1302	LG4 Statistical	Sum Average (d = 1, 135°)	1	1	1	1	1	1	1	1
1303	LG4 Statistical	Sum Entropy (d = 1, 135°)	1	0	0	0	1	0	0	0
1304	LG4 Statistical	Sum Variance (d = 1, 135°)	1	0	0	0	1	0	0	0
1305	LG4 Statistical	Difference Entropy (d = 1, 135°)	1	0	0	0	0	0	0	0
1306	LG4 Statistical	Difference Variance (d = 1, 135°)	1	0	0	0	0	0	0	0
1307	LG4 Statistical	Homogeneity 1 (d = 1, 135°)	1	0	0	0	0	0	0	0
1308	LG4 Statistical	Homogeneity 2 (d = 1, 135°)	1	0	0	0	0	0	0	0
1309	LG4 Statistical	Energy (d = 3, 0°)	1	0	0	0	1	0	0	0
1310	LG4 Statistical	Variance (d = 3, 0°)	1	1	1	1	1	1	1	1
1311	LG4 Statistical	Contrast (d = 3, 0°)	0	0	0	0	0	0	0	0
1312	LG4 Statistical	Dissimilarity (d = 3, 0°)	0	0	0	0	0	0	0	0
1313	LG4 Statistical	Correlation (d = 3, 0°)	1	1	1	1	1	1	1	1
1314	LG4 Statistical	Autocorrelation (d = 3, 0°)	1	1	1	1	1	1	1	1
1315	LG4 Statistical	Entropy (d = 3, 0°)	1	0	0	0	0	0	0	0
1316	LG4 Statistical	Measure of Correlation 1 (d = 3, 0°)	0	0	0	0	0	0	0	0
1317	LG4 Statistical	Measure of Correlation 2 (d = 3, 0°)	1	1	1	1	1	1	1	1
1318	LG4 Statistical	Cluster Shade (d = 3, 0°)	1	1	1	1	1	1	1	1
1319	LG4 Statistical	Cluster Prominence (d = 3, 0°)	1	1	1	1	1	1	1	1

1320	LG4 Statistical	Max Probability (d = 3, 0°)	1	0	0	0	1	0	0	0
1321	LG4 Statistical	Sum Average (d = 3, 0°)	1	1	1	1	1	1	1	1
1322	LG4 Statistical	Sum Entropy (d = 3, 0°)	1	0	0	0	0	0	0	0
1323	LG4 Statistical	Sum Variance (d = 3, 0°)	1	0	0	0	0	0	0	0
1324	LG4 Statistical	Difference Entropy (d = 3, 0°)	0	0	0	0	0	0	0	0
1325	LG4 Statistical	Difference Variance (d = 3, 0°)	0	0	0	0	0	0	0	0
1326	LG4 Statistical	Homogeneity 1 (d = 3, 0°)	0	0	0	0	0	0	0	0
1327	LG4 Statistical	Homogeneity 2 (d = 3, 0°)	0	0	0	0	0	0	0	0
1328	LG4 Statistical	Energy (d = 3, 45°)	1	0	0	0	1	0	0	0
1329	LG4 Statistical	Variance (d = 3, 45°)	1	1	1	1	1	1	1	1
1330	LG4 Statistical	Contrast (d = 3, 45°)	0	0	0	0	0	0	0	0
1331	LG4 Statistical	Dissimilarity (d = 3, 45°)	0	0	0	0	0	0	0	0
1332	LG4 Statistical	Correlation (d = 3, 45°)	1	1	1	1	1	1	1	1
1333	LG4 Statistical	Autocorrelation (d = 3, 45°)	1	1	1	1	1	1	1	1
1334	LG4 Statistical	Entropy (d = 3, 45°)	1	0	0	0	0	0	0	0
1335	LG4 Statistical	Measure of Correlation 1 (d = 3, 45°)	0	0	0	0	0	0	0	0
1336	LG4 Statistical	Measure of Correlation 2 (d = 3, 45°)	1	1	1	1	1	1	1	1
1337	LG4 Statistical	Cluster Shade (d = 3, 45°)	1	1	1	1	1	1	1	1
1338	LG4 Statistical	Cluster Prominence (d = 3, 45°)	1	1	1	1	1	1	1	1
1339	LG4 Statistical	Max Probability (d = 3, 45°)	1	0	0	0	1	0	0	0
1340	LG4 Statistical	Sum Average (d = 3, 45°)	1	1	1	1	1	1	1	1
1341	LG4 Statistical	Sum Entropy (d = 3, 45°)	1	0	0	0	0	0	0	0
1342	LG4 Statistical	Sum Variance (d = 3, 45°)	1	0	0	0	0	0	0	0
1343	LG4 Statistical	Difference Entropy (d = 3, 45°)	0	0	0	0	0	0	0	0
1344	LG4 Statistical	Difference Variance (d = 3, 45°)	0	0	0	0	0	0	0	0
1345	LG4 Statistical	Homogeneity 1 (d = 3, 45°)	0	0	0	0	0	0	0	0
1346	LG4 Statistical	Homogeneity 2 (d = 3, 45°)	0	0	0	0	0	0	0	0
1347	LG4 Statistical	Energy (d = 3, 90°)	1	0	0	0	1	0	0	0
1348	LG4 Statistical	Variance (d = 3, 90°)	1	1	1	1	1	1	1	1
1349	LG4 Statistical	Contrast (d = 3, 90°)	0	0	0	0	0	0	0	0

1350	LG4 Statistical	Dissimilarity (d = 3, 90°)	0	0	0	0	0	0	0	0
1351	LG4 Statistical	Correlation (d = 3, 90°)	1	1	1	0	0	0	0	0
1352	LG4 Statistical	Autocorrelation (d = 3, 90°)	1	1	1	1	1	1	1	1
1353	LG4 Statistical	Entropy (d = 3, 90°)	1	0	0	0	0	0	0	0
1354	LG4 Statistical	Measure of Correlation 1 (d = 3, 90°)	0	0	0	0	0	0	0	0
1355	LG4 Statistical	Measure of Correlation 2 (d = 3, 90°)	1	1	1	1	1	1	1	1
1356	LG4 Statistical	Cluster Shade (d = 3, 90°)	1	1	1	1	1	1	1	1
1357	LG4 Statistical	Cluster Prominence (d = 3, 90°)	1	1	1	1	1	1	1	1
1358	LG4 Statistical	Max Probability (d = 3, 90°)	1	0	0	0	1	0	0	0
1359	LG4 Statistical	Sum Average (d = 3, 90°)	1	1	1	1	1	1	1	1
1360	LG4 Statistical	Sum Entropy (d = 3, 90°)	1	0	0	0	1	0	0	0
1361	LG4 Statistical	Sum Variance (d = 3, 90°)	1	0	0	0	1	0	0	0
1362	LG4 Statistical	Difference Entropy (d = 3, 90°)	0	0	0	0	0	0	0	0
1363	LG4 Statistical	Difference Variance (d = 3, 90°)	0	0	0	0	0	0	0	0
1364	LG4 Statistical	Homogeneity 1 (d = 3, 90°)	0	0	0	0	0	0	0	0
1365	LG4 Statistical	Homogeneity 2 (d = 3, 90°)	0	0	0	0	0	0	0	0
1366	LG4 Statistical	Energy (d = 3, 135°)	1	0	0	0	1	0	0	0
1367	LG4 Statistical	Variance (d = 3, 135°)	1	1	1	1	1	1	1	1
1368	LG4 Statistical	Contrast (d = 3, 135°)	0	0	0	0	0	0	0	0
1369	LG4 Statistical	Dissimilarity (d = 3, 135°)	0	0	0	0	0	0	0	0
1370	LG4 Statistical	Correlation (d = 3, 135°)	1	1	1	0	1	0	0	0
1371	LG4 Statistical	Autocorrelation (d = 3, 135°)	1	1	1	1	1	1	1	1
1372	LG4 Statistical	Entropy (d = 3, 135°)	1	0	0	0	0	0	0	0
1373	LG4 Statistical	Measure of Correlation 1 (d = 3, 135°)	0	0	0	0	0	0	0	0
1374	LG4 Statistical	Measure of Correlation 2 (d = 3, 135°)	1	1	1	1	1	1	1	1
1375	LG4 Statistical	Cluster Shade (d = 3, 135°)	1	1	1	1	1	1	1	1
1376	LG4 Statistical	Cluster Prominence (d = 3, 135°)	1	1	1	1	1	1	1	1
1377	LG4 Statistical	Max Probability (d = 3, 135°)	1	0	0	0	1	0	0	0
1378	LG4 Statistical	Sum Average (d = 3, 135°)	1	1	1	1	1	1	1	1
1379	LG4 Statistical	Sum Entropy (d = 3, 135°)	1	0	0	0	0	0	0	0

1380	LG4 Statistical	Sum Variance (d = 3, 135°)	1	0	0	0	0	0	0	0
1381	LG4 Statistical	Difference Entropy (d = 3, 135°)	0	0	0	0	0	0	0	0
1382	LG4 Statistical	Difference Variance (d = 3, 135°)	0	0	0	0	0	0	0	0
1383	LG4 Statistical	Homogeneity 1 (d = 3, 135°)	0	0	0	0	0	0	0	0
1384	LG4 Statistical	Homogeneity 2 (d = 3, 135°)	0	0	0	0	0	0	0	0
1385	LG4 Statistical	Energy (d = 5, 0°)	1	0	0	0	1	0	0	0
1386	LG4 Statistical	Variance (d = 5, 0°)	1	1	1	1	1	1	1	1
1387	LG4 Statistical	Contrast (d = 5, 0°)	0	0	0	0	0	0	0	0
1388	LG4 Statistical	Dissimilarity (d = 5, 0°)	0	0	0	0	0	0	0	0
1389	LG4 Statistical	Correlation (d = 5, 0°)	1	1	1	1	1	1	1	1
1390	LG4 Statistical	Autocorrelation (d = 5, 0°)	1	1	1	1	1	1	1	1
1391	LG4 Statistical	Entropy (d = 5, 0°)	1	0	0	0	0	0	0	0
1392	LG4 Statistical	Measure of Correlation 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
1393	LG4 Statistical	Measure of Correlation 2 (d = 5, 0°)	1	1	1	1	1	1	1	1
1394	LG4 Statistical	Cluster Shade (d = 5, 0°)	1	1	1	1	1	1	1	1
1395	LG4 Statistical	Cluster Prominence (d = 5, 0°)	1	1	1	1	1	1	1	1
1396	LG4 Statistical	Max Probability (d = 5, 0°)	1	0	0	0	1	0	0	0
1397	LG4 Statistical	Sum Average (d = 5, 0°)	1	1	1	1	1	1	1	1
1398	LG4 Statistical	Sum Entropy (d = 5, 0°)	1	0	0	0	0	0	0	0
1399	LG4 Statistical	Sum Variance (d = 5, 0°)	1	0	0	0	0	0	0	0
1400	LG4 Statistical	Difference Entropy (d = 5, 0°)	0	0	0	0	0	0	0	0
1401	LG4 Statistical	Difference Variance (d = 5, 0°)	0	0	0	0	0	0	0	0
1402	LG4 Statistical	Homogeneity 1 (d = 5, 0°)	0	0	0	0	0	0	0	0
1403	LG4 Statistical	Homogeneity 2 (d = 5, 0°)	0	0	0	0	0	0	0	0
1404	LG4 Statistical	Energy (d = 5, 45°)	1	0	0	0	0	0	0	0
1405	LG4 Statistical	Variance (d = 5, 45°)	1	1	1	1	1	1	1	1
1406	LG4 Statistical	Contrast (d = 5, 45°)	0	0	0	0	0	0	0	0
1407	LG4 Statistical	Dissimilarity (d = 5, 45°)	0	0	0	0	0	0	0	0
1408	LG4 Statistical	Correlation (d = 5, 45°)	1	1	1	1	1	1	1	1
1409	LG4 Statistical	Autocorrelation (d = 5, 45°)	1	1	1	1	1	1	1	1

1410	LG4 Statistical	Entropy (d = 5, 45°)	1	0	0	0	0	0	0	0
1411	LG4 Statistical	Measure of Correlation 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
1412	LG4 Statistical	Measure of Correlation 2 (d = 5, 45°)	1	1	1	1	1	1	1	1
1413	LG4 Statistical	Cluster Shade (d = 5, 45°)	1	1	1	1	1	1	1	1
1414	LG4 Statistical	Cluster Prominence (d = 5, 45°)	1	1	1	1	1	1	1	1
1415	LG4 Statistical	Max Probability (d = 5, 45°)	1	0	0	0	1	0	0	0
1416	LG4 Statistical	Sum Average (d = 5, 45°)	1	1	1	1	1	1	1	1
1417	LG4 Statistical	Sum Entropy (d = 5, 45°)	1	0	0	0	0	0	0	0
1418	LG4 Statistical	Sum Variance (d = 5, 45°)	1	0	0	0	0	0	0	0
1419	LG4 Statistical	Difference Entropy (d = 5, 45°)	0	0	0	0	0	0	0	0
1420	LG4 Statistical	Difference Variance (d = 5, 45°)	0	0	0	0	0	0	0	0
1421	LG4 Statistical	Homogeneity 1 (d = 5, 45°)	0	0	0	0	0	0	0	0
1422	LG4 Statistical	Homogeneity 2 (d = 5, 45°)	0	0	0	0	0	0	0	0
1423	LG4 Statistical	Energy (d = 5, 90°)	1	0	0	0	1	0	0	0
1424	LG4 Statistical	Variance (d = 5, 90°)	1	1	1	1	1	1	1	1
1425	LG4 Statistical	Contrast (d = 5, 90°)	0	0	0	0	0	0	0	0
1426	LG4 Statistical	Dissimilarity (d = 5, 90°)	0	0	0	0	0	0	0	0
1427	LG4 Statistical	Correlation (d = 5, 90°)	1	1	1	1	1	1	1	1
1428	LG4 Statistical	Autocorrelation (d = 5, 90°)	1	1	1	1	1	1	1	1
1429	LG4 Statistical	Entropy (d = 5, 90°)	1	0	0	0	0	0	0	0
1430	LG4 Statistical	Measure of Correlation 1 (d = 5, 90°)	0	0	0	0	0	0	0	0
1431	LG4 Statistical	Measure of Correlation 2 (d = 5, 90°)	1	1	1	1	1	1	1	1
1432	LG4 Statistical	Cluster Shade (d = 5, 90°)	1	1	1	1	1	1	1	1
1433	LG4 Statistical	Cluster Prominence (d = 5, 90°)	1	1	1	1	1	1	1	1
1434	LG4 Statistical	Max Probability (d = 5, 90°)	1	0	0	0	1	0	0	0
1435	LG4 Statistical	Sum Average (d = 5, 90°)	1	1	1	1	1	1	1	1
1436	LG4 Statistical	Sum Entropy (d = 5, 90°)	1	0	0	0	0	0	0	0
1437	LG4 Statistical	Sum Variance (d = 5, 90°)	1	0	0	0	0	0	0	0
1438	LG4 Statistical	Difference Entropy (d = 5, 90°)	0	0	0	0	0	0	0	0
1439	LG4 Statistical	Difference Variance (d = 5, 90°)	0	0	0	0	0	0	0	0

1440	LG4 Statistical	Homogeneity 1 (d = 5, 90°)	0	0	0	0	0	0	0	0
1441	LG4 Statistical	Homogeneity 2 (d = 5, 90°)	0	0	0	0	0	0	0	0
1442	LG4 Statistical	Energy (d = 5, 135°)	1	0	0	0	1	0	0	0
1443	LG4 Statistical	Variance (d = 5, 135°)	1	1	1	1	1	1	1	1
1444	LG4 Statistical	Contrast (d = 5, 135°)	0	0	0	0	0	0	0	0
1445	LG4 Statistical	Dissimilarity (d = 5, 135°)	0	0	0	0	0	0	0	0
1446	LG4 Statistical	Correlation (d = 5, 135°)	1	1	1	1	1	1	1	1
1447	LG4 Statistical	Autocorrelation (d = 5, 135°)	1	1	1	1	1	1	1	1
1448	LG4 Statistical	Entropy (d = 5, 135°)	1	0	0	0	0	0	0	0
1449	LG4 Statistical	Measure of Correlation 1 (d = 5, 135°)	0	0	0	0	0	0	0	0
1450	LG4 Statistical	Measure of Correlation 2 (d = 5, 135°)	1	1	1	1	1	1	1	1
1451	LG4 Statistical	Cluster Shade (d = 5, 135°)	1	1	1	1	1	1	1	1
1452	LG4 Statistical	Cluster Prominence (d = 5, 135°)	1	1	1	1	1	1	1	1
1453	LG4 Statistical	Max Probability (d = 5, 135°)	1	0	0	0	1	0	0	0
1454	LG4 Statistical	Sum Average (d = 5, 135°)	1	1	1	1	1	1	1	1
1455	LG4 Statistical	Sum Entropy (d = 5, 135°)	1	0	0	0	0	0	0	0
1456	LG4 Statistical	Sum Variance (d = 5, 135°)	1	0	0	0	0	0	0	0
1457	LG4 Statistical	Difference Entropy (d = 5, 135°)	0	0	0	0	0	0	0	0
1458	LG4 Statistical	Difference Variance (d = 5, 135°)	0	0	0	0	0	0	0	0
1459	LG4 Statistical	Homogeneity 1 (d = 5, 135°)	0	0	0	0	0	0	0	0
1460	LG4 Statistical	Homogeneity 2 (d = 5, 135°)	0	0	0	0	0	0	0	0

Appendix C

Appendix C - Difference between HOG Entropy and EHHOG

HOG (histogram of oriented gradients) entropy is an existing feature used in [117] for image compensation. HOG entropy and our proposed feature, EHHOG (entropy of histogram of HOG), have similar names but are 2 different features. Their differences are summarized in Table C.1 below.

TABLE C.1: Differences between HOG (histogram of oriented gradients) entropy and our proposed EHHOG (entropy of histogram of HOG)

Feature	Whether can pass boxplot check for our dataset	Rationale	HOG formation	Calculation
HOG entropy	0	Minimum entropy of HOG corresponds to maximum number of straight lines (verified in 123 images out of 285 in [117])	1 cell = 1 block = whole image	$-\sum_{all\ h} h(\log_2 h)$ h denotes the values of HOG vector
EHHOG	1	EHHOG represents whether an image has more "no edge" cells or more "edge" cells	1 cell = 8*8 pixels 1 block = 2*2 cells, with 50% overlapping between blocks	$-\sum_{all\ p} p(\log_2 p)$ p denotes the values of histogram of HOG vector, i.e., the probabilities of h

In HOG formation, the orientations of image gradients are discretized into 9 bins: $\theta_1 = [0^\circ, 20^\circ)$, $\theta_2 = [20^\circ, 40^\circ)$, $\theta_3 = [40^\circ, 60^\circ)$, $\theta_4 = [60^\circ, 80^\circ)$, $\theta_5 = [80^\circ, 100^\circ)$, $\theta_6 = [100^\circ, 120^\circ)$, $\theta_7 = [120^\circ, 140^\circ)$, $\theta_8 = [140^\circ, 160^\circ)$, $\theta_9 = [160^\circ, 180^\circ)$. In [117], they did not divide the image into cells and blocks (i.e., 1 cell = 1 block = whole image), so the image HOG will be a 1*9 vector, corresponding to the 9 orientation bins.

In the proposed feature EHHOG, we used the classical HOG formation, i.e., an image is divided into 8*8 pixel cells, then the cells form 2*2 cell blocks with 50% overlapping [99]. For example, in Table C.2, for HOG calculation, the 1st block has cells 1, 4, 2, 5; the 2nd block has cells 2, 5, 3, 6; the 3rd block has cells 4, 7, 5, 8; the 4th block has cells 5, 8, 6, 9. Then

the HOG for the whole image will be formed by the HOGs of each cell as in Figure C.1. For our 488×648 pixel images, the number of blocks is $(488/8 - 1) * (648/8 - 1) = 4800$, each block has $2 * 2 = 4$ cells, each cell HOG has 9 orientation bins, and $4800 * 4 * 9 = 172800$, so the image HOG will be a $1 * 172800$ vector.

TABLE C.2: Illustration of image cells for HOG (histogram of oriented gradients)

Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6
Cell 7	Cell 8	Cell 9

FIGURE C.1: Demonstration of HOG (histogram of oriented gradients) formation used for our feature EHHOG (entropy of histogram of HOG)

Appendix D

Appendix D - One or Two Repetitions of 10-Fold Cross-Validation

10-fold cross-validation (CV) is the training-testing partition technique used in our experiments. With this technique, the classifier/ model will be trained and tested for 10 times and the average values will be the final results. If 2 repetitions of 10-fold CV were done, there will be 20 times of training and testing.

TABLE D.1: Classification results with 1 or 2 repetitions of 10-fold CV using RBF SVM as the classifier on our dataset

CV: cross-validation. RBF: radial basis function. SVM: support vector machine. * denotes the proposed method. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. BiGHAM method: using feature combination $G1B1B\alpha$ with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination $G1B1ARSO$ with circle-based Multi-Region Binarization (Chapter 6). acc. refers to accuracy. std. dev. refers to standard deviation. 1 R, 2 R refer to 1 repetition, 2 repetitions of 10-fold CV.

Number of images per class	BiGHAM *				iBiGHAM *			
	Test acc. mean		Test acc. std. dev.		Test acc. mean		Test acc. std. dev.	
	1 R	2 R	1 R	2 R	1 R	2 R	1 R	2 R
30	77.8%	80.0%	11.7%	12.3%	85.6%	86.7%	11.8%	11.2%
70	86.7%	88.1%	7.0%	7.3%	87.6%	90.5%	6.4%	6.2%
100	92.3%	92.5%	4.2%	4.7%	92.0%	90.8%	6.5%	5.7%
300	94.0%	96.8%	2.1%	1.9%	95.8%	98.1%	2.1%	1.4%
700	94.3%	94.3%	1.7%	1.5%	97.1%	96.8%	1.4%	1.1%
1000	94.8%	95.0%	1.3%	1.1%	96.6%	96.7%	1.5%	1.1%

TABLE D.2: Classification results with 1 or 2 repetitions of 10-fold CV using BDT as the classifier on our dataset

CV: cross-validation. BDT: bagging decision trees. * denotes the proposed method. BiGHAM: Binarized-Greyscale-Hybrid Algorithm with Multi-Region Binarization. iBiGHAM: improved-BiGHAM. BiGHAM method: using feature combination *G1B1B α* with rectangle-based Multi-Region Binarization (Chapter 5). iBiGHAM method: using feature combination *G1B1ARSOCO* with circle-based Multi-Region Binarization (Chapter 6). acc. refers to accuracy. std. dev. refers to standard deviation. 1 R, 2 R refer to 1 repetition, 2 repetitions of 10-fold CV.

Number of images per class	BiGHAM *				iBiGHAM *			
	Test acc. mean		Test acc. std. dev.		Test acc. mean		Test acc. std. dev.	
	1 R	2 R	1 R	2 R	1 R	2 R	1 R	2 R
30	82.2%	83.4%	11.9%	12.2%	81.1%	82.2%	10.5%	12.7%
70	90.5%	91.2%	6.7%	6.6%	90.0%	90.5%	6.1%	5.6%
100	91.3%	90.5%	4.5%	5.7%	95.3%	94.0%	4.2%	4.3%
300	95.8%	96.9%	1.6%	1.8%	97.0%	97.4%	2.0%	2.3%
700	98.3%	98.1%	0.6%	0.9%	98.6%	98.7%	0.6%	0.6%
1000	98.7%	98.8%	1.0%	0.7%	99.2%	99.2%	0.5%	0.5%

From Tables D.1, D.2, it can be observed that from 1 repetition to 2 repetitions of 10-fold cross-validation (CV), the testing accuracies did not have significant differences at all number of images in terms of both mean values and standard deviations. Thus, for our classification experiments in the main text, only 1 repetition of 10-fold CV was used to save time and computing resources.

Appendix E

Appendix E - Experiment Results of Deep Learning

The deep-learning-based Pedraza's method [73] based on the pre-trained AlexNet [72] model was applied to our dataset and the open-sourced dataset BCO-DMO 670706 [23] with a batch size of 100 samples. Apart from the original setting in the paper [73] which did not perform very well in our experiments (Tables E.1, E.5), 36 settings were implemented. Three dropout values were used, 30%, 50%, 70%, for the fully-connected (FC) layer. There were 12 learning rates (LRs):

- 1) Fixed LR, 0.05
- 2) Fixed LR, 0.01
- 3) Fixed LR, 0.005
- 4) Fixed LR, 0.001
- 5) Step LR, initial at 0.05, decrease to 0.8 of the previous LR with every 10 epochs
- 6) Step LR, initial at 0.05, decrease to 0.8 of the previous LR with every 20 epochs
- 7) Step LR, initial at 0.01, decrease to 0.8 of the previous LR with every 10 epochs
- 8) Step LR, initial at 0.01, decrease to 0.8 of the previous LR with every 20 epochs
- 9) Step LR, initial at 0.005, decrease to 0.8 of the previous LR with every 10 epochs
- 10) Step LR, initial at 0.005, decrease to 0.8 of the previous LR with every 20 epochs
- 11) Step LR, initial at 0.001, decrease to 0.8 of the previous LR with every 10 epochs

12) Step LR, initial at 0.001, decrease to 0.8 of the previous LR with every 20 epochs

Among these settings, the setting "dropout = 50%, step learning rate (LR) initialized at 0.001, decreased to 0.8 of the previous LR with every 10 epochs (LR 11)" was used when comparing the proposed methods and Pedraza's method on our dataset for it performed well at all conditions of numbers of samples (images) per class.

Among these settings, the setting "dropout = 30%, fixed LR initialized at 0.005 (LR 3), symmetric padding for small datasets (less than 100 images per class), grey padding for other cases" was used when comparing the proposed methods and Pedraza's method on the open-sourced dataset BCO-DMO 670706 [23] for it performed well at all conditions of numbers of samples (images) per class.

With the limitation of time and computing resources, the following settings were fixed: using stochastic gradient descent (SGD) for optimization during training, using 100 as the batch size, and training and testing for hundreds of epochs then picking the accuracy at the epoch where the accuracy was stable and not improving after that epoch (on the accuracy-epoch plot). For example, in Figure E.1, the testing accuracy became stable at 300 epochs, so the accuracy 92.4% was recorded in Table E.6 for the LR 3, 1000-image-per-class case.

FIGURE E.1: Example of accuracy-epoch plot

Using the DL-based method [73] to classify the images in the open-sourced dataset [23], grey padding for data augmentation, 30% dropout, fixed learning rate (LR) at 0.005, 1000 images per class

E.1 Experiment Results of Pedraza’s method on Our Dataset

TABLE E.1: Testing accuracies of using the DL-based method [73] to classify our images, original setting

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 147, 117, 148 images for class G, C, O respectively. Original setting, the same setting as in the paper [73]: 50% dropout, step learning rate (LR), initial at 0.001, decrease to 0.9 of previous LR with every 8 epochs.

Num. of images per class	Testing accuracy
Ori.	91.3%
30	84.4%
70	89.5%
100	91.7%
300	96.4%
700	97.2%
1000	98.8%

TABLE E.2: Testing accuracies of using the DL-based method [73] to classify our images, 30% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 147, 117, 148 images for class G, C, O respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	35.7%	45.0%	93.4%	94.9%	35.7%	35.7%	47.6%	39.8%	84.9%	93.0%	93.2%	91.3%
30	33.3%	67.8%	74.4%	83.3%	33.3%	33.3%	81.1%	75.6%	76.7%	83.3%	80.0%	83.3%
70	33.3%	34.3%	81.9%	89.5%	33.3%	33.3%	60.0%	63.3%	81.9%	88.6%	90.5%	90.5%
100	33.3%	40.0%	82.3%	95.7%	33.3%	33.3%	39.7%	33.3%	83.7%	93.7%	94.7%	95.3%
300	33.3%	52.8%	87.6%	96.0%	33.3%	33.3%	79.2%	86.4%	86.3%	95.6%	96.3%	96.3%
700	33.3%	60.0%	90.5%	98.5%	33.3%	33.3%	39.8%	40.0%	85.0%	98.5%	96.0%	96.5%
1000	33.3%	40.0%	93.4%	98.8%	33.3%	33.3%	33.3%	46.7%	86.6%	86.6%	98.8%	98.8%

TABLE E.3: Testing accuracies of using the DL-based method [73] to classify our images, 50% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 147, 117, 148 images for class G, C, O respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	35.7%	53.7%	91.7%	93.7%	35.7%	35.7%	35.7%	35.7%	90.6%	92.5%	93.9%	90.0%
30	33.3%	82.2%	85.6%	82.2%	33.3%	33.3%	83.3%	73.3%	81.1%	82.2%	83.3%	82.2%
70	33.3%	46.7%	85.7%	89.5%	33.3%	33.3%	60.5%	39.5%	89.5%	86.2%	92.4%	91.0%
100	33.3%	33.3%	95.0%	95.3%	33.3%	33.3%	33.3%	34.3%	92.3%	91.3%	95.7%	95.0%
300	33.3%	91.3%	95.8%	95.7%	33.3%	33.3%	75.8%	66.4%	92.7%	92.8%	96.2%	96.2%
700	33.3%	46.7%	97.8%	97.9%	33.3%	33.3%	40.0%	46.7%	95.9%	94.8%	96.6%	96.1%
1000	33.3%	33.3%	99.6%	99.0%	33.3%	33.3%	33.3%	40.0%	98.8%	86.7%	98.8%	98.9%

TABLE E.4: Testing accuracies of using the DL-based method [73] to classify our images, 70% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 147, 117, 148 images for class G, C, O respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	35.7%	48.1%	93.5%	91.7%	35.7%	35.7%	46.4%	35.4%	90.3%	90.5%	93.2%	88.3%
30	33.3%	78.9%	86.7%	82.2%	33.3%	33.3%	84.4%	82.2%	77.8%	84.4%	84.4%	83.3%
70	33.3%	54.3%	89.5%	89.1%	33.3%	33.3%	49.1%	66.7%	82.9%	87.1%	91.4%	87.6%
100	33.3%	41.0%	91.7%	94.3%	33.3%	33.3%	48.7%	39.3%	88.0%	94.7%	96.3%	94.3%
300	33.3%	72.0%	95.6%	95.2%	33.3%	33.3%	89.7%	79.2%	89.2%	96.0%	95.7%	95.8%
700	33.3%	53.2%	97.9%	98.3%	33.3%	33.3%	46.7%	46.6%	93.1%	97.6%	96.5%	96.1%
1000	33.3%	33.3%	99.2%	98.7%	33.3%	33.3%	40.0%	40.0%	93.2%	98.8%	98.8%	98.8%

FIGURE E.2: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 1

DL-based: deep-learning-based.

FIGURE E.3: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 2

DL-based: deep-learning-based.

FIGURE E.4: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 3

DL-based: deep-learning-based

FIGURE E.5: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 4

DL-based: deep-learning-based.

FIGURE E.6: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 5

DL-based: deep-learning-based.

FIGURE E.7: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 6

DL-based: deep-learning-based.

FIGURE E.8: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 7

DL-based: deep-learning-based.

FIGURE E.9: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 8

DL-based: deep-learning-based.

FIGURE E.10: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 9

DL-based: deep-learning-based.

FIGURE E.11: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 10

DL-based: deep-learning-based.

FIGURE E.12: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 11

DL-based: deep-learning-based.

FIGURE E.13: Testing accuracies of using the DL-based method [73] to classify our images with different dropout values, LR 12

DL-based: deep-learning-based.

E.2 Experiment Results of Pedraza’s method on Open-Sourced Dataset BCO-DMO 670706

TABLE E.5: Training and testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], original setting

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively. Original setting, the same setting as in the paper [73]: 50% dropout, step learning rate (LR), initial at 0.001, decrease to 0.9 of previous LR with every 8 epochs.

Num. of images per class	Testing accuracy	
	Grey padding	Symmetric padding
Ori.	100%	100%
30	33.3%	80.0%
70	50.0%	80.7%
100	60.5%	55.0%
300	64.3%	69.3%
700	76.1%	71.6%
1000	78.8%	73.9%

TABLE E.6: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], grey padding for data augmentation, 30% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%
30	51.7%	51.7%	51.7%	55.0%	53.3%	50.0%	48.3%	51.7%	41.7%	51.7%	35.0%	38.3%
70	50.0%	52.1%	52.9%	58.6%	50.0%	50.0%	52.1%	52.9%	53.6%	52.1%	50.0%	52.9%
100	50.0%	73.5%	82.5%	82.0%	50.0%	50.0%	51.0%	75.5%	61.0%	70.5%	64.5%	71.0%
300	50.0%	76.8%	86.0%	85.3%	50.0%	50.0%	52.2%	76.3%	55.0%	73.2%	67.7%	75.0%
700	50.0%	83.6%	89.3%	91.7%	50.0%	50.0%	90.2%	90.9%	89.9%	90.4%	87.7%	86.8%
1000	50.0%	92.1%	92.4%	92.4%	50.0%	50.0%	91.2%	92.0%	90.9%	91.7%	89.9%	91.1%

TABLE E.7: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], grey padding for data augmentation, 50% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	50.0%	50.0%	95.0%	100%	50.0%	50.0%	50.0%	60.0%	100%	100%	50.0%	50.0%
30	50.0%	53.3%	51.7%	31.7%	51.7%	50.0%	50.0%	50.0%	50.0%	50.0%	38.3%	35.0%
70	50.0%	57.1%	53.6%	54.3%	53.6%	52.9%	50.7%	54.3%	50.7%	51.4%	50.7%	51.4%
100	50.0%	68.0%	61.5%	64.5%	68.0%	69.0%	68.0%	65.5%	64.5%	69.5%	65.5%	70.5%
300	50.0%	70.2%	56.3%	57.5%	69.2%	70.5%	69.0%	70.3%	68.3%	70.2%	65.8%	72.2%
700	50.0%	91.6%	78.0%	89.9%	77.7%	80.5%	78.1%	80.2%	77.5%	81.9%	87.6%	86.5%
1000	50.0%	92.6%	50.3%	59.2%	80.2%	81.5%	79.9%	82.0%	79.5%	82.1%	89.0%	91.1%

TABLE E.8: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], grey padding for data augmentation, 70% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%
30	45.0%	50.0%	50.0%	51.7%	45.0%	50.0%	40.0%	51.7%	43.3%	51.7%	35.0%	36.7%
70	50.0%	52.9%	51.4%	60.0%	50.0%	50.0%	52.1%	52.1%	50.7%	52.9%	50.7%	52.9%
100	50.0%	69.5%	80.5%	82.0%	50.0%	50.0%	63.0%	65.0%	59.0%	72.5%	64.5%	69.5%
300	50.0%	71.0%	83.0%	85.3%	50.0%	50.0%	55.0%	68.0%	53.0%	74.5%	65.5%	70.3%
700	50.0%	91.4%	88.8%	91.7%	50.0%	50.0%	86.6%	88.7%	88.1%	90.2%	85.7%	85.1%
1000	50.0%	93.1%	92.9%	92.4%	50.0%	50.0%	90.4%	92.4%	90.1%	92.1%	89.1%	91.1%

TABLE E.9: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], symmetric padding for data augmentation, 30% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%
30	50.0%	81.7%	80.0%	80.0%	50.0%	50.0%	80.0%	80.0%	80.0%	80.0%	78.3%	78.3%
70	50.0%	75.7%	81.4%	80.0%	50.0%	50.0%	75.0%	81.4%	80.0%	80.7%	80.0%	80.7%
100	50.0%	54.5%	54.0%	55.5%	50.0%	50.0%	54.5%	53.3%	55.0%	54.5%	59.0%	56.5%
300	50.0%	64.0%	65.3%	66.5%	50.0%	50.0%	68.0%	67.0%	68.3%	67.0%	68.5%	68.8%
700	50.0%	76.9%	75.3%	72.5%	50.0%	50.0%	76.1%	72.7%	74.4%	78.0%	75.4%	73.7%
1000	50.0%	79.6%	80.5%	78.6%	50.0%	50.0%	78.5%	79.7%	78.6%	79.5%	77.9%	78.3%

TABLE E.10: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], symmetric padding for data augmentation, 50% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	50.0%	50.0%	95.0%	100%	50.0%	50.0%	50.0%	60.0%	100%	100%	50.0%	50.0%
30	53.3%	80.0%	80.0%	78.3%	75.0%	78.3%	78.3%	80.0%	80.0%	80.0%	80.0%	78.3%
70	50.0%	81.4%	80.0%	80.0%	80.0%	80.7%	81.4%	80.0%	80.7%	80.0%	80.7%	80.0%
100	50.0%	53.5%	55.0%	54.0%	61.5%	61.5%	61.0%	60.0%	57.5%	62.0%	57.0%	57.5%
300	50.0%	64.8%	62.7%	64.7%	67.8%	67.3%	67.3%	67.7%	68.0%	67.8%	68.8%	68.0%
700	50.0%	80.3%	52.0%	72.8%	68.6%	68.0%	70.6%	70.6%	71.0%	72.6%	75.5%	73.4%
1000	50.0%	80.6%	50.1%	54.8%	71.0%	70.4%	72.6%	73.9%	73.3%	74.0%	77.8%	78.1%

TABLE E.11

TABLE E.12: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23], symmetric padding for data augmentation, 70% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 14, 6 images for type 1, 2 respectively.

Num. of images per class	Learning rate (LR)											
	1	2	3	4	5	6	7	8	9	10	11	12
Ori.	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%
30	53.3%	76.7%	80.0%	78.3%	51.7%	55.0%	80.0%	80.0%	81.7%	80.0%	78.3%	80.0%
70	50.0%	80.7%	80.7%	80.0%	50.0%	50.0%	78.6%	81.4%	80.0%	80.7%	80.0%	80.0%
100	50.0%	57.5%	54.5%	55.0%	50.0%	50.0%	53.5%	53.5%	55.5%	54.0%	57.5%	56.5%
300	50.0%	65.3%	64.8%	66.7%	50.0%	50.0%	66.0%	67.0%	68.0%	68.0%	68.5%	68.3%
700	50.0%	72.2%	77.2%	74.2%	50.0%	50.0%	74.1%	76.9%	77.8%	78.3%	76.1%	72.8%
1000	50.0%	80.4%	80.2%	79.1%	50.0%	50.0%	79.1%	79.3%	78.6%	79.1%	77.9%	78.0%

FIGURE E.14: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 1

DL-based: deep-learning-based.

FIGURE E.15: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 2

DL-based: deep-learning-based.

FIGURE E.16: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 3

DL-based: deep-learning-based.

FIGURE E.17: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 4

DL-based: deep-learning-based.

FIGURE E.18: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 5

DL-based: deep-learning-based.

FIGURE E.19: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 6

DL-based: deep-learning-based.

FIGURE E.20: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 7

DL-based: deep-learning-based.

FIGURE E.21: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 8

DL-based: deep-learning-based.

FIGURE E.22: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 9

DL-based: deep-learning-based.

FIGURE E.23: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 10

DL-based: deep-learning-based.

FIGURE E.24: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 11

DL-based: deep-learning-based.

FIGURE E.25: Testing accuracies of using the DL-based method [73] to classify the images in the open-sourced dataset [23] with different dropout values and padding methods, LR 12

DL-based: deep-learning-based.

E.3 Experiment Results of MobileNetV2 on Our Dataset

MobileNetV2 [104] is a deep learning (DL) model proposed and designed for mobile applications. Its standard (input image size: 224*224 pixels) computational complexity is 300 MFLOPs, less than half of Pedraza's method (716 MFLOPs), but still much higher than the handcrafted-feature-based proposed algorithms (about 40 MFLOPs). This DL model was applied to our dataset but with bad classification performances as shown in Table E.13. Due to the limitation of time and computing resources, only the learning rates (LRs) 1, 2, 3, 4 described before and the original LR in the paper [104] with 50% dropout was tried.

In Table E.13, it can be observed that MobileNetV2 occasionally achieved high accuracies, but for most of the time, the classification performances were not good and without regularity. For future studies, more settings for MobileNetV2 can be tried and similar lightweight DL models can be investigated.

TABLE E.13: Testing accuracies of using the DL-based MobileNetV2 [104] to classify the images in our dataset, 50% dropout

DL-based: deep-learning-based. Ori. refers to the original dataset without data augmentation, it has 147, 117, 148 images for class G, C, O respectively. Original learning rate (LR), the same LR as in the paper [104]: step learning rate (LR), initial at 0.045, decrease to 0.98 of previous LR with every epoch.

Num. of images per class	Learning rate (LR)				
	LR 1	LR 2	LR 3	LR 4	Original LR
Ori.	54.1%	80.3%	97.8%	80.3%	59.2%
30	90.0%	97.8%	97.8%	97.8%	97.8%
70	59.1%	98.6%	98.6%	98.6%	80.0%
100	39.7%	85.3%	91.3%	85.3%	44.3%
300	39.1%	60.0%	69.1%	60.0%	35.2%
700	33.4%	43.5%	37.9%	43.5%	34.0%
1000	33.3%	34.6%	33.5%	34.6%	33.8%

Appendix F

Appendix F - Calculation Method of Aspect Ratio, Solidity, and Compactness

The aspect ratio (AR) calculation requires the major axis length and the minor axis length. The solidity (SO) calculation requires the convex hull of an object. The compactness (CO) calculation requires the area and perimeter of the object. The program is implemented in MATLAB.

The major (minor) axis length of an object refers to the length (in pixels) of the major (minor) axis of the ellipse that has the same normalized 2nd central moments as the object. The major (minor) axis of an ellipse is shown in Figure F.1. An object and the ellipse with the same 2nd central moment is shown in Figure F.2. The 2nd order central moment of an image $f(x, y)$ can be calculated as Equation (F.1).

$$\mu_{11} = \frac{\sum_{all\ x} \sum_{all\ y} (x - \bar{x})(y - \bar{y})f(x, y)}{\sum_{all\ x} \sum_{all\ y} f(x, y)} \quad (F.1)$$

Here, $(\bar{x}, \bar{y})$ denotes the centroid location of $f(x, y)$.

The AR, defined as the ratio of the major axis length and the minor axis length, is calculated by Equation (F.2).

$$AR = \frac{L_{major\ axis}}{L_{minor\ axis}} \quad (F.2)$$

FIGURE F.1: Illustration of major axis and minor axis

Image source: https://en.wikipedia.org/wiki/Aspect_ratio

FIGURE F.2: Illustration of an object and the ellipse with the same 2nd central moment (left), the ellipse with major axis and minor axis (right)

Image source: <http://www.mathworks.com>

The convex hull of an object is the smallest convex set that contains the object. Some objects (displayed as original) with the corresponding convex hulls are shown in Figure F.3 for illustration.

The SO, defined as the ratio of the object's area and the area of the convex hull of the object, is calculated by Equation (F.3).

$$SO = \frac{A_{object}}{A_{convex\ hull}} \quad (F.3)$$

FIGURE F.3: Illustration of convex hull of an object

Image source: <http://resources.esri.com>

The CO, defined as the ratio of an object's area and the area of a circle that has the same perimeter of the object, is calculated by Equation (F.4).

$$\begin{aligned}
 CO &= \frac{A_{object}}{A_{circle}} \\
 &= \frac{A_{object}}{\pi \left(\frac{P_{object}}{2\pi} \right)^2} \\
 &= \frac{4\pi A_{object}}{P_{object}^2}
 \end{aligned} \tag{F.4}$$

Here, A_{object} denotes the area of the object, P_{object} denotes the perimeter of the object.

Appendix G

Appendix G - Computational Complexity Calculation Method and Example

G.1 Introduction

The number of FLOPs (floating-point operations) is a common metric to evaluate the computational complexity of an algorithm, especially frequently used for deep learning (DL) models [118]. It is also called the number of Multiply-Adds with Mult-Adds or MAdds for short. The formulas to calculate the number of FLOPs for a convolutional neural network (CNN) in a forward pass (i.e., testing one image) are as follows.

$$\text{Conv. layer} : N_{conv} = (C_{in}K^2 + 1) H_{out} W_{out} C_{out} \quad (\text{G.1})$$

$$\text{FC layer} : N_{FC} = (I - 1)O \quad (\text{G.2})$$

Here, N_{conv} denotes the number of FLOPs for a convolutional layer, C_{in} denotes the number of channels of the input feature map, K denotes the kernel size, and H_{out} , W_{out} , C_{out} denotes the height, width, number of channels of the output feature map. N_{FC} denotes the number of FLOPs for an FC (fully-connected) layer, I denotes the number of the input neurons, and O denotes the number of the output neurons.

The units MFLOPs and GFLOPs are also commonly used as below.

$$1 \text{ MFLOPs} = 1 \times 10^6 \text{ FLOPs} = 1 \text{ million FLOPs} \quad (\text{G.3})$$

$$1 \text{ GFLOPs} = 1 \times 10^9 \text{ FLOPs} = 1 \text{ billion FLOPs} \quad (\text{G.4})$$

The computational complexity in terms of the number of FLOPs is approximately proportional to the total number of pixels in the image, i.e., the multiplication result of image height and image width.

G.2 Calculation Example

Using the famous DL CNN model, AlexNet [72], as an example, the steps to calculate the number of FLOPs are demonstrated in this section. The CNN structure of AlexNet is shown in Figure G.1.

FIGURE G.1: Illustration of the structure of AlexNet [72]

Image source: Paper "Imagenet classification with deep convolutional neural networks" [72]

In the experiments, the AlexNet was implemented using Python language with the *pytorch* framework. The FLOP calculation steps of the AlexNet model in *pytorch* are demonstrated in Table G.1.

TABLE G.1: Layer-by-layer demonstration of computational complexity calculation of AlexNet [72]

FLOP: floating-point operation, multiply-accumulate operation considered, $G: \times 10^9$. Computational complexity for a forward pass (i.e., testing one image) in terms of number of FLOPs. Conv: convolutional layer. Max pool: max pooling layer, down-sampling the images by using 1 pixel with the maximum among the 4 pixel values to represent each 2×2 pixel regions. FC: fully-connected layer. In Layer 1, the 3-channel 224×224 input image was filtered by 96 11×11 convolutional kernels with a stride of 4 pixels, so the output image size/resolution is 55×55 . The pooling and ReLU (Rectified Linear Unit) are ignored in the calculation since their computational complexities are very small compared to the convolutional layers and FC layers.

Number	Input size	Layer type	Output size	Calculation	Computational complexity
1	$224 \times 224, 3$	11×11 Conv, 64	$55 \times 55, 64$	$(3 \times 11^2 + 1) \times 55 \times 55 \times 64$	70470400 FLOPs $\approx$ 0.070 GFLOPs
2	$55 \times 55, 64$	Max pool	$27 \times 27, 64$	Nil	Nil
3	$27 \times 27, 64$	5×5 Conv, 192	$27 \times 27, 192$	$(64 \times 5^2 + 1) \times 27 \times 27 \times 192$	224088768 FLOPs $\approx$ 0.224 GFLOPs
4	$27 \times 27, 192$	Max pool	$13 \times 13, 192$	Nil	Nil
5	$13 \times 13, 192$	3×3 Conv, 384	$13 \times 13, 384$	$(192 \times 3^2 + 1) \times 13 \times 13 \times 384$	112205184 FLOPs $\approx$ 0.112 GFLOPs
6	$13 \times 13, 384$	3×3 Conv, 256	$13 \times 13, 256$	$(384 \times 3^2 + 1) \times 13 \times 13 \times 256$	149563648 FLOPs $\approx$ 0.150 GFLOPs
7	$13 \times 13, 256$	3×3 Conv, 256	$13 \times 13, 256$	$(256 \times 3^2 + 1) \times 13 \times 13 \times 256$	99723520 FLOPs $\approx$ 0.100 GFLOPs
8	$13 \times 13, 256$	Max pool	$6 \times 6, 256$	Nil	Nil
9	$6 \times 6 \times 256 =$ 9216 neurons	FC	4096 neurons	$(9216 - 1) \times 4096$	37744640 FLOPs $\approx$ 0.038 GFLOPs
10	4096 neurons	FC	4096 neurons	$(4096 - 1) \times 4096$	16773120 FLOPs $\approx$ 0.017GFLOPs
11	4096 neurons	FC	1000 neurons	$(4096 - 1) \times 1000$	4095000 FLOPs $\approx$ 0.004 GFLOPs
Summation of all the layers					714664280 GFLOPs $\approx$ 0.715 GFLOPs

G.3 Open-Sourced Tools

The number of FLOPs can be calculated with open-sourced tools available online.

For DL CNNs that were implemented using Python language, the number of FLOPs was calculated with the package *ptflops* [119] in the experiments. The screenshot of running *ptflops* to calculate the computational complexity of AlexNet are shown in Figure G.3. It should be noted that the input image size for the pre-trained AlexNet model is 224*224 pixels, and our image size is 488*648 pixels (Section 3.1), so a pre-processing for adjusting the image size must be applied before using the pre-trained AlexNet model to classify our image. Specifically, our images were resized to 224*297 pixels and then centre cropped to 224*224 pixels. The computational complexity is 0.715 GFLOPs to test one 224*224 image in Table G.1, and 0.716 GFLOPs in Figure G.3. The minor difference is due to the FLOPs of the pooling and ReLU (rectified linear unit) calculations in the CNN neglected in Table G.1. In the thesis, 0.716 GFLOPs, i.e., 716 MFLOPs, were used.

```

import torchvision.models as models
from ptflops import get_model_complexity_info
netAlex = models.alexnet()
FLOPs, Params = get_model_complexity_info(netAlex, (3, 224, 224), as_strings=True, print_per_layer_stat=True)
AlexNet(
  0.716 GMac, 100.000% MACs,
  (features): Sequential(
    0.657 GMac, 91.805% MACs,
    (0): Conv2d(0.07 GMac, 9.848% MACs, 3, 64, kernel_size=(11, 11), stride=(4, 4), padding=(2, 2))
    (1): ReLU(0.0 GMac, 0.027% MACs, inplace)
    (2): MaxPool2d(0.0 GMac, 0.027% MACs, kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)
    (3): Conv2d(0.224 GMac, 31.317% MACs, 64, 192, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (4): ReLU(0.0 GMac, 0.020% MACs, inplace)
    (5): MaxPool2d(0.0 GMac, 0.020% MACs, kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)
    (6): Conv2d(0.112 GMac, 15.681% MACs, 192, 384, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (7): ReLU(0.0 GMac, 0.009% MACs, inplace)
    (8): Conv2d(0.15 GMac, 20.902% MACs, 384, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (9): ReLU(0.0 GMac, 0.006% MACs, inplace)
    (10): Conv2d(0.1 GMac, 13.937% MACs, 256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (11): ReLU(0.0 GMac, 0.006% MACs, inplace)
    (12): MaxPool2d(0.0 GMac, 0.006% MACs, kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (avgpool): AdaptiveAvgPool2d(0.0 GMac, 0.001% MACs, output_size=(6, 6))
  (classifier): Sequential(
    0.059 GMac, 8.194% MACs,
    (0): Dropout(0.0 GMac, 0.000% MACs, p=0.5)
    (1): Linear(0.038 GMac, 5.275% MACs, in_features=9216, out_features=4096, bias=True)
    (2): ReLU(0.0 GMac, 0.001% MACs, inplace)
    (3): Dropout(0.0 GMac, 0.000% MACs, p=0.5)
    (4): Linear(0.017 GMac, 2.345% MACs, in_features=4096, out_features=4096, bias=True)
    (5): ReLU(0.0 GMac, 0.001% MACs, inplace)
    (6): Linear(0.004 GMac, 0.572% MACs, in_features=4096, out_features=1000, bias=True)
  )
)

```

FIGURE G.2

FIGURE G.3: Calculating the number of FLOPs of AlexNet [72] with Python package *ptflops* [119]

FLOP: floating-point operation, multiply-accumulate operation considered.

For non-CNN algorithms that were implemented using MATLAB language, the number of FLOPs was calculated with the tool "FLOPS" [120] in the experiments.

Appendix H

Appendix H - Examples of Rotated Images and Corresponding Features

In the experiments, the data augmentation was done by image rotation. Below illustrates some image examples and the corresponding feature values with 2-degree, 4-degree, ..., 24-degree rotations. It can be observed that the features vary with different rotation degrees, i.e., the features are not rotation invariant. Although the morphological features (AR, SO, CO) are theoretically rotation invariant for the same object, the change of the ratio of the un-rotated original intensities and padding areas made the binarized object different, which made the features different. Thus, image rotation is a simple and effective way to augment data.

H.1 Image Example of Our Dataset

FIGURE H.1: Examples of rotated images of class O, # 85 image of our dataset

TABLE H.1: Feature values extracted from rotated images of class O, # 85 image of our dataset with rectangle-based Multi-Region Binarization

G1 refers to G1_EHHOG, as there is only 1 feature, EHHOG, in the feature set G1.

Feature	Rotation degree											
	2	4	6	8	10	12	14	16	18	20	22	24
G1	6.126	6.112	6.084	6.054	6.013	5.981	5.95	5.919	5.887	5.868	5.839	5.827
B1_AR	2.44	1.32	2.44	2.425	2.438	1.318	2.425	1.36	1.314	2.437	2.44	1.318
B1_SO	0.54	0.6	0.541	0.545	0.541	0.599	0.545	0.569	0.6	0.541	0.541	0.6
B1_CO	0.043	0.068	0.041	0.039	0.043	0.071	0.038	0.063	0.066	0.04	0.04	0.072
B2_AR	1.571	1.55	1.55	1.557	1.58	1.577	1.5	1.571	1.577	1.484	1.567	1.559
B2_SO	0.604	0.618	0.619	0.62	0.605	0.604	0.596	0.606	0.603	0.57	0.582	0.585
B2_CO	0.038	0.042	0.042	0.042	0.036	0.036	0.028	0.037	0.036	0.028	0.034	0.034
B3_AR	1.537	1.581	1.588	1.596	1.604	1.707	1.714	1.719	1.719	1.726	1.731	1.731
B3_SO	0.605	0.595	0.59	0.591	0.592	0.618	0.62	0.623	0.625	0.624	0.626	0.624
B3_CO	0.034	0.033	0.033	0.031	0.031	0.042	0.042	0.043	0.041	0.042	0.042	0.042
B4_AR	1.618	1.532	1.634	1.532	1.53	1.531	1.533	1.537	1.539	1.542	1.552	1.544
B4_SO	0.637	0.555	0.644	0.544	0.566	0.568	0.567	0.569	0.553	0.551	0.569	0.555
B4_CO	0.032	0.025	0.041	0.026	0.027	0.026	0.027	0.025	0.028	0.032	0.03	0.032
B5_AR	1.533	1.532	1.531	1.512	1.529	1.526	1.524	1.304	1.295	1.317	1.317	1.27
B5_SO	0.57	0.582	0.599	0.571	0.575	0.576	0.584	0.546	0.546	0.545	0.551	0.556
B5_CO	0.026	0.031	0.027	0.029	0.025	0.026	0.023	0.021	0.023	0.02	0.019	0.025
B6_AR	1.099	1.14	1.144	1.157	1.165	1.188	1.182	1.194	1.202	1.204	1.207	1.21
B6_SO	0.551	0.539	0.548	0.557	0.558	0.57	0.567	0.578	0.584	0.585	0.592	0.584
B6_CO	0.016	0.01	0.01	0.015	0.012	0.013	0.022	0.021	0.022	0.021	0.021	0.02

TABLE H.2: Feature values extracted from rotated images of class O, # 85 image of our dataset with circle-based Multi-Region Binarization

G1 refers to G1_EHHOG, as there is only 1 feature, EHHOG, in the feature set G1.

Feature	Rotation degree											
	2	4	6	8	10	12	14	16	18	20	22	24
G1	6.126	6.112	6.084	6.054	6.013	5.981	5.95	5.919	5.887	5.868	5.839	5.827
B1_AR	2.44	1.32	2.44	2.425	2.438	1.318	2.425	1.36	1.314	2.437	2.44	1.318
B1_SO	0.54	0.6	0.541	0.545	0.541	0.599	0.545	0.569	0.6	0.541	0.541	0.6
B1_CO	0.043	0.068	0.041	0.039	0.043	0.071	0.038	0.063	0.066	0.04	0.04	0.072
B2_AR	1.506	1.302	1.505	1.506	1.508	1.299	1.426	1.302	1.305	1.427	1.507	1.303
B2_SO	0.523	0.586	0.524	0.523	0.523	0.587	0.537	0.585	0.582	0.536	0.524	0.581
B2_CO	0.026	0.028	0.026	0.027	0.026	0.027	0.022	0.026	0.026	0.021	0.025	0.025
B3_AR	1.512	1.482	1.301	1.229	1.637	1.481	1.229	1.396	1.394	1.228	1.229	1.392
B3_SO	0.573	0.617	0.523	0.524	0.624	0.616	0.523	0.595	0.594	0.524	0.523	0.591
B3_CO	0.023	0.031	0.019	0.015	0.029	0.031	0.016	0.021	0.023	0.015	0.016	0.023
B4_AR	1.599	1.236	1.707	1.285	1.281	1.209	1.265	1.218	1.479	1.273	1.263	1.212
B4_SO	0.618	0.602	0.618	0.548	0.551	0.618	0.595	0.612	0.655	0.554	0.597	0.612
B4_CO	0.023	0.017	0.031	0.018	0.019	0.019	0.017	0.017	0.031	0.019	0.015	0.018
B5_AR	1.292	1.143	1.306	1.311	1.316	1.169	1.326	1.176	1.181	1.335	1.337	1.192
B5_SO	0.596	0.59	0.601	0.604	0.608	0.594	0.608	0.598	0.604	0.613	0.614	0.609
B5_CO	0.012	0.017	0.013	0.013	0.013	0.016	0.013	0.017	0.016	0.014	0.013	0.019
B6_AR	1.357	1.189	1.373	1.379	1.384	1.214	1.393	1.22	1.233	1.403	1.407	1.239
B6_SO	0.665	0.643	0.673	0.677	0.681	0.659	0.687	0.664	0.694	0.696	0.7	0.699
B6_CO	0.027	0.022	0.027	0.027	0.029	0.026	0.03	0.026	0.029	0.033	0.038	0.032

H.2 Image Example of Open-Sourced Dataset BCO-DMO 670706

FIGURE H.2: Examples of rotated images of type 1, # 1 image of the open-sourced dataset BCO-DMO 670706 [23], grey padding

TABLE H.3: Feature values extracted from rotated images of type 1, # 1 image of the open-sourced dataset BCO-DMO 670706 [23] with grey padding, rectangle-based Multi-Region Binarization

G1 refers to G1_EHHOG, as there is only 1 feature, EHHOG, in the feature set G1.

Feature	Rotation degree											
	2	4	6	8	10	12	14	16	18	20	22	24
G1	5.578	5.582	5.562	5.538	5.516	5.488	5.461	5.431	5.406	5.379	5.366	5.353
B1_AR	1.133	1.134	1.141	1.141	1.142	1.14	1.14	1.135	1.138	1.138	1.137	1.136
B1_SO	0.813	0.821	0.813	0.806	0.808	0.811	0.81	0.8	0.798	0.799	0.8	0.799
B1_CO	0.121	0.118	0.112	0.118	0.135	0.121	0.139	0.149	0.138	0.149	0.135	0.136
B2_AR	1.133	1.134	1.141	1.141	1.142	1.14	1.14	1.135	1.138	1.138	1.137	1.136
B2_SO	0.813	0.821	0.813	0.806	0.808	0.811	0.81	0.8	0.798	0.799	0.8	0.799
B2_CO	0.121	0.118	0.112	0.118	0.135	0.121	0.139	0.149	0.138	0.149	0.135	0.136
B3_AR	1.133	1.134	1.141	1.141	1.142	1.14	1.14	1.135	1.138	1.138	1.137	1.136
B3_SO	0.813	0.821	0.813	0.806	0.808	0.811	0.81	0.8	0.798	0.799	0.8	0.799
B3_CO	0.121	0.118	0.112	0.118	0.135	0.121	0.139	0.149	0.138	0.149	0.135	0.136
B4_AR	1.129	1.131	1.137	1.138	1.139	1.136	1.135	1.13	1.131	1.13	1.129	1.125
B4_SO	0.833	0.838	0.833	0.831	0.841	0.843	0.841	0.846	0.849	0.847	0.845	0.844
B4_CO	0.138	0.137	0.134	0.137	0.136	0.144	0.172	0.197	0.19	0.197	0.172	0.177
B5_AR	1.093	1.092	1.09	1.086	1.087	1.082	1.081	1.077	1.08	1.076	1.072	1.068
B5_SO	0.891	0.896	0.904	0.9	0.896	0.908	0.912	0.924	0.919	0.926	0.928	0.921
B5_CO	0.347	0.325	0.344	0.347	0.368	0.382	0.365	0.434	0.492	0.5	0.537	0.501
B6_AR	1.055	1.051	1.055	1.051	1.056	1.061	1.06	1.057	1.066	1.06	1.063	1.069
B6_SO	0.84	0.847	0.849	0.852	0.86	0.856	0.867	0.867	0.861	0.863	0.867	0.858
B6_CO	0.299	0.312	0.3	0.284	0.297	0.267	0.276	0.265	0.252	0.264	0.252	0.235

TABLE H.4: Feature values extracted from rotated images of type 1, # 1 image of the open-sourced dataset BCO-DMO 670706 [23] with grey padding, circle-based Multi-Region Binarization

G1 refers to G1_EHHOG, as there is only 1 feature, EHHOG, in the feature set G1.

Feature	Rotation degree											
	2	4	6	8	10	12	14	16	18	20	22	24
G1	5.578	5.582	5.562	5.538	5.516	5.488	5.461	5.431	5.406	5.379	5.366	5.353
B1_AR	1.133	1.134	1.141	1.141	1.142	1.14	1.14	1.135	1.138	1.138	1.137	1.136
B1_SO	0.813	0.821	0.813	0.806	0.808	0.811	0.81	0.8	0.798	0.799	0.8	0.799
B1_CO	0.121	0.118	0.112	0.118	0.135	0.121	0.139	0.149	0.138	0.149	0.135	0.136
B2_AR	1.133	1.134	1.141	1.141	1.142	1.14	1.14	1.135	1.138	1.138	1.137	1.136
B2_SO	0.813	0.821	0.813	0.806	0.808	0.811	0.81	0.8	0.798	0.799	0.8	0.799
B2_CO	0.121	0.118	0.112	0.118	0.135	0.121	0.139	0.149	0.138	0.149	0.135	0.136
B3_AR	1.133	1.134	1.141	1.141	1.142	1.14	1.14	1.135	1.138	1.138	1.137	1.136
B3_SO	0.813	0.821	0.813	0.806	0.808	0.811	0.81	0.8	0.798	0.799	0.8	0.799
B3_CO	0.121	0.118	0.112	0.118	0.135	0.121	0.139	0.149	0.138	0.149	0.135	0.136
B4_AR	1.103	1.104	1.104	1.098	1.098	1.093	1.09	1.086	1.086	1.086	1.085	1.082
B4_SO	0.856	0.861	0.867	0.871	0.878	0.882	0.889	0.898	0.903	0.904	0.904	0.906
B4_CO	0.176	0.169	0.176	0.215	0.221	0.206	0.233	0.267	0.293	0.302	0.299	0.293
B5_AR	1.03	1.032	1.034	1.036	1.035	1.038	1.038	1.038	1.04	1.04	1.041	1.042
B5_SO	0.919	0.921	0.927	0.926	0.923	0.921	0.915	0.916	0.914	0.915	0.914	0.912
B5_CO	0.581	0.531	0.543	0.519	0.528	0.525	0.503	0.511	0.495	0.488	0.474	0.428
B6_AR	1.048	1.05	1.055	1.054	1.053	1.061	1.059	1.062	1.07	1.069	1.067	1.074
B6_SO	0.825	0.829	0.833	0.831	0.839	0.839	0.853	0.85	0.839	0.836	0.84	0.844
B6_CO	0.267	0.267	0.25	0.236	0.24	0.237	0.257	0.237	0.2	0.207	0.196	0.197

FIGURE H.3: Examples of rotated images of type 1, # 1 image of the open-sourced dataset BCO-DMO 670706 [23], symmetric padding

TABLE H.5: Feature values extracted from rotated images of type 1, # 1 image of the open-sourced dataset BCO-DMO 670706 [23] with symmetric padding, rectangle-based Multi-Region Binarization

G1 refers to G1_EHHOG, as there is only 1 feature, EHHOG, in the feature set G1.

Feature	Rotation degree											
	2	4	6	8	10	12	14	16	18	20	22	24
G1	5.517	5.511	5.495	5.486	5.48	5.47	5.464	5.454	5.45	5.443	5.448	5.454
B1_AR	1.668	1.656	2.875	2.892	2.854	2.879	2.833	2.838	2.829	2.779	2.64	2.687
B1_SO	0.482	0.481	0.524	0.518	0.512	0.506	0.512	0.508	0.502	0.499	0.477	0.493
B1_CO	0.005	0.004	0.008	0.006	0.006	0.007	0.008	0.008	0.008	0.007	0.006	0.006
B2_AR	1.621	1.645	1.654	1.665	1.637	1.64	1.642	1.631	1.615	1.633	1.613	1.585
B2_SO	0.65	0.642	0.646	0.646	0.639	0.634	0.645	0.657	0.649	0.647	0.638	0.641
B2_CO	0.013	0.015	0.015	0.014	0.011	0.013	0.014	0.017	0.016	0.017	0.014	0.012
B3_AR	1.254	1.263	1.265	1.272	1.266	1.269	1.272	1.261	1.252	1.262	1.262	1.244
B3_SO	0.802	0.8	0.804	0.804	0.807	0.805	0.808	0.804	0.806	0.804	0.797	0.794
B3_CO	0.041	0.049	0.047	0.046	0.047	0.037	0.044	0.046	0.044	0.044	0.04	0.036
B4_AR	1.094	1.09	1.092	1.093	1.098	1.102	1.093	1.091	1.086	1.085	1.085	1.078
B4_SO	0.922	0.922	0.926	0.929	0.924	0.928	0.934	0.939	0.937	0.933	0.934	0.942
B4_CO	0.215	0.177	0.22	0.192	0.215	0.212	0.213	0.21	0.194	0.209	0.201	0.207
B5_AR	1.053	1.051	1.049	1.047	1.045	1.049	1.046	1.046	1.048	1.053	1.055	1.066
B5_SO	0.867	0.872	0.866	0.862	0.865	0.869	0.876	0.865	0.869	0.861	0.861	0.86
B5_CO	0.105	0.107	0.11	0.101	0.098	0.094	0.112	0.103	0.1	0.103	0.091	0.088
B6_AR	1.12	1.121	1.119	1.119	1.121	1.122	1.124	1.123	1.124	1.129	1.124	1.126
B6_SO	0.832	0.83	0.836	0.834	0.829	0.83	0.83	0.83	0.829	0.833	0.83	0.831
B6_CO	0.065	0.064	0.061	0.058	0.058	0.063	0.054	0.056	0.056	0.052	0.059	0.055

TABLE H.6: Feature values extracted from rotated images of type 1, # 1 image of the open-sourced dataset BCO-DMO 670706 [23] with symmetric padding, circle-based Multi-Region Binarization

G1 refers to G1_EHHOG, as there is only 1 feature, EHHOG, in the feature set G1.

Feature	Rotation degree											
	2	4	6	8	10	12	14	16	18	20	22	24
G1	5.517	5.511	5.495	5.486	5.48	5.47	5.464	5.454	5.45	5.443	5.448	5.454
B1_AR	1.668	1.656	2.875	2.892	2.854	2.879	2.833	2.838	2.829	2.779	2.64	2.687
B1_SO	0.482	0.481	0.524	0.518	0.512	0.506	0.512	0.508	0.502	0.499	0.477	0.493
B1_CO	0.005	0.004	0.008	0.006	0.006	0.007	0.008	0.008	0.008	0.007	0.006	0.006
B2_AR	1.433	1.434	1.432	1.435	1.416	1.427	1.425	1.426	1.418	1.444	1.433	1.412
B2_SO	0.656	0.654	0.658	0.656	0.651	0.651	0.663	0.678	0.669	0.67	0.66	0.66
B2_CO	0.016	0.016	0.016	0.016	0.013	0.014	0.019	0.019	0.019	0.02	0.017	0.013
B3_AR	1.121	1.122	1.122	1.121	1.121	1.122	1.116	1.117	1.112	1.113	1.118	1.115
B3_SO	0.855	0.855	0.855	0.854	0.855	0.854	0.861	0.86	0.862	0.861	0.861	0.862
B3_CO	0.082	0.087	0.085	0.085	0.082	0.077	0.087	0.081	0.087	0.083	0.086	0.083
B4_AR	1.011	1.011	1.011	1.011	1.011	1.011	1.01	1.01	1.01	1.01	1.01	1.01
B4_SO	0.976	0.975	0.975	0.975	0.976	0.976	0.975	0.975	0.974	0.975	0.977	0.975
B4_CO	0.589	0.567	0.547	0.543	0.568	0.573	0.543	0.536	0.542	0.533	0.561	0.55
B5_AR	1.116	1.111	1.116	1.114	1.113	1.11	1.112	1.112	1.108	1.117	1.108	1.111
B5_SO	0.843	0.847	0.843	0.841	0.845	0.846	0.848	0.84	0.844	0.834	0.842	0.842
B5_CO	0.058	0.052	0.054	0.05	0.053	0.051	0.064	0.062	0.056	0.06	0.054	0.055
B6_AR	1.131	1.132	1.13	1.131	1.132	1.131	1.135	1.132	1.131	1.135	1.13	1.131
B6_SO	0.824	0.824	0.829	0.827	0.823	0.823	0.825	0.824	0.824	0.828	0.825	0.826
B6_CO	0.061	0.06	0.057	0.055	0.055	0.06	0.052	0.053	0.055	0.051	0.057	0.053