Advances in Detection and Classification of Underwater Targets using Synthetic Aperture Sonar Imagery

Vom Fachbereich Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von
Dipl.-Ing. Tai Fei
geboren am 27.01.1983 in Suzhou, Jiangsu, V.R. China

Referent:	Prof. Dr.-Ing. Abdelhak M. Zoubir
Korreferenten:	Prof. Dr.-Ing. Dieter Kraus
	Prof. Dr.-Ing. Marius Pesavento
Tag der Einreichung:	03.06.2014
Tag der mündlichen Prüfung:	28.11.2014

D17
Darmstadt, 2015

Acknowledgments

I am very grateful to my doctoral mentor Prof. Dr.-Ing. Abdelhak M. Zoubir for his supervision. It is truly a pleasure being supervised by an outstanding researcher who shows such a high degree of enthusiasm and motivation. Prof. Zoubir provided me with an inspiring mixture of freedom in research and guidance which made my time as a PhD student a pleasure. I wish to thank Prof. Dr.-Ing. Dieter Kraus for his supervision, guidance and numerous technical discussions during the past more than three years in Bremen. I benefited greatly from our interactions in both research and normal life, and I am delighted to have such a renowned and compassionate researcher as my co-advisor.

I would also like to give my acknowledgment to the Ministry of Economics and Technology of the German Federal Government for their funding of my work in the project of CImaging, and ATLAS Elektronik Bremen GmbH for their assistance with the sonar data.

I also want to thank Prof. K. Hofmann, Prof. Dr.-Ing. Marius Pesavento, Prof. Dr.-Ing. F. Küppers and Prof. Dr.-Ing. A. Schürr who acted as chair and examiners in the PhD committee.

My thanks go to my colleagues at the Institute of Water Acoustics, Sonar Engineering and Signal Theory at Hochschule Bremen, and the Signal Processing Group at Technische Universität Darmstadt. I was very happy to work in such a convivial environment. Thanks to Ange Francine Tchinda Pockem, Benjamin Lehmann, Martin Walzer, Ivan Aleksi, Christian Debes, Sara Al-Sayed, Mouhammad Alhumaidi, Nevine Demitri, Michael Fauss, Gökhan Gül, Jürgen Hahn, Philipp Heidenreich, Sahar Khawatmi, Stefan Leier, Michael Leigsnering, Zhihua Lu, Ahmed A. Mostafa, Michael Muma, Waqas Sharif, Fiky Y. Suratman, Gebremichael Teame, Christian Weiss, Feng Yin, as well as Rene Ramson, Renate Koschella and Hauke Fath.

I wish to thank my parents Yongming Fei and Jun Zhu for their unconditional love and support throughout my life. I would also like to thank all my loved ones for your selfless support, especially my uncle Mr. Jerry Tan, who is a sincere friend of my father and provided me with much precious advice to help me conquer all the challenges on the way of my life. Finally, I am most grateful to my lovely wife Yifei Guo for her understanding, love, encouragement, support and joy.

Bremen, May 25, 2014

Kurzfassung

In dieser Doktorarbeit wird das Problem von Detektion und Klassifizierung der Unterwasserminen auf Sonarbildern betrachtet. Die automatische Erkennung und automatische Klassifizierung (automatic detection and automatic classification, ADAC) wird auf Bilder angewandt, die mit Hilfe des synthetischen Apertur-Sonars (SAS) entstanden sind. Das ADAC-System besteht aus vier Bereichen: Detektion minenähnlicher Objekte, Bildsegmentierung, Extraktion der Merkmale und Klassifizierung der Minen. Diese Doktorarbeit konzentriert sich auf die letzten drei Bereiche.

Bei der Detektion minenähnlicher Objekte (mine-like object, MLO) wird die Template-Matching-Technik auf die Sonarbilder angewandt. Diese Technik basiert auf der A-priori-Kenntnis der Minenformen. Damit sind die Bereiche mit den MLO festgelegt. Diese Bereiche werden Bereiche von Interesse genannt (regions of interest, ROI). Die ROI werden von den Sonarbildern extrahiert und an die zwei folgenden Module, d.h. Bildsegmentierung und Extraktion der Merkmale, übermittelt.

Bei der Bildsegmentierung wird eine modifizierte Erwartungsmaximierung zur Segmentierung der Bilder vorgeschlagen. Zwecks Klassifizierung der MLO-Formen werden die Sonarbilder in Objekt, Objektschatten und Hintergrund aufgeteilt. Ein allgemeines Mischmodell wird für die statistische Auswertung der Bilddaten eingesetzt. Außerdem wird eine Clusterung der Bildpunkte im Rahmen der Dempster-Shafer-Theorie (DST) verwendet, um die räumliche Abhängigkeit zwischen den Bildpunkten zu berücksichtigen. Folglich werden die Störflecke im Hintergrundbereich beseitigt. Optimale Konfigurationen für diesen Ansatz werden mit Hilfe quantitativer numerischer Studien ermittelt.

Die extrahierten Merkmale werden an das Klassifizierungsmodul weitergegeben. Berücksichtigt werden vor allem geometrische und Textur-Merkmale. In der Literatur werden zahlreiche Merkmale vorgeschlagen, die die Objektform und die Textur beschreiben können.

Aufgrund des Fluches der Dimensionalität ist die Merkmalsauswahl unerlässlich für die Entwicklung eines ADAC-Systems. Eine anspruchsvolle Filter-Methode zur Selektierung optimaler Merkmale für die Objektklassifikation wird entwickelt. Diese Filter-Methode benutzt ein neuartiges Gütemaß zur Beurteilung der Relevanz von Merkmalen. Das Gütemaß ist eine Kombination aus gegenseitigen Informationen, dem modifizierten Relief-Gewicht und der Shannon-Entropie. Die ausgewählten Merkmale zeigen eine höhere Generalisierbarkeit auf. Im Vergleich zu anderen Methoden führen die nach

der hier vorgeschlagenen Methode ausgesuchten Merkmale zu einer sehr guten Klassifizierungsgüte, und die Performance-Abweichung bei Verwendung unterschiedlicher Klassifikatoren nimmt ab.

Bei der Minen-Klassifizierung wird die Voraussage der Typen Minenähnlicher Objekte betrachtet. Ein Kombinationsschema auf Grundlage der DST wird vorgeschlagen, das die einander ergänzenden Informationen unterschiedlicher Klassifikatoren nutzt. Die Ergebnisse einzelner Klassifikatoren werden mit Hilfe des entwickelten Schemas kombiniert. Die resultierende Klassifikationsgenauigkeit ist höher als die von jedem einzelnen Klassifikator.

Alle erwähnten Methoden werden anhand der SAS-Bilder evaluiert. Abschließend wird ein Fazit gezogen und einige Anregungen für zukünftige Arbeiten werden gegeben.

Abstract

In this PhD thesis, the problem of underwater mine detection and classification using synthetic aperture sonar (SAS) imagery is considered. The automatic detection and automatic classification (ADAC) system is applied to images obtained by SAS systems. The ADAC system contains four steps, namely mine-like object (MLO) detection, image segmentation, feature extraction, and mine type classification. This thesis focuses on the last three steps.

In the mine-like object detection step, a template-matching technique based on the *a priori* knowledge of mine shapes is applied to scan the sonar imagery for the detection of MLOs. Regions containing MLOs are called regions of interest (ROI). They are extracted and forwarded to the subsequent steps, i.e. image segmentation and feature extraction.

In the image segmentation step, a modified expectation-maximization (EM) approach is proposed. For the sake of acquiring the shape information of the MLO in the ROI, the SAS images are segmented into highlights, shadows, and backgrounds. A generalized mixture model is adopted to approximate the statistics of the image data. In addition, a Dempster-Shafer theory-based clustering technique is used to consider the spatial correlation between pixels so that the clutters in background regions can be removed. Optimal parameter settings for the proposed EM approach are found with the help of quantitative numerical studies.

In the feature extraction step, features are extracted and will be used as the inputs for the mine type classification step. Both the geometrical features and the texture features are applied. However, there are numerous features proposed to describe the object shape and the texture in the literature.

Due to the curse of dimensionality, it is indispensable to do the feature selection during the design of an ADAC system. A sophisticated filter method is developed to choose optimal features for the classification purpose. This filter method utilizes a novel feature relevance measure that is a combination of the mutual information, the modified Relief weight, and the Shannon entropy. The selected features demonstrate a higher generalizability. Compared with other filter methods, the features selected by our method can lead to superior classification accuracy, and their performance variation over different classifiers is decreased.

In the mine type classification step, the prediction of the types of MLO is considered. In order to take advantage of the complementary information among different classifiers,

a classifier combination scheme is developed in the framework of the Dempster-Shafer theory. The outputs of individual classifiers are combined according to this classifier combination scheme. The resulting classification accuracy is better than those of individual classifiers.

All of the proposed methods are evaluated using SAS data. Finally, conclusions are drawn, and some suggestions about future works are proposed as well.

Contents

Chapter 1

Motivation and Introduction

The basic process of noticing an object and recognizing what it is happens frequently in our daily life. The ease with which we deal with these issues belies the astoundingly complex processing in our brains. Over the past tens of millions of years, a highly sophisticated neural and cognitive system has evolved for us to tackle such issues. Nowadays, thanks to the rapid development of high-performance computers, automatic target recognition (ATR) [1] becomes possible. It has numerous civilian and military applications, such as face recognition [2, 3], medical application [4] and target recognition using radar signals [5–14].

This thesis deals with ATR in the underwater application using sonar imagery. Compared with the imagery acquired by digital cameras or radar systems, the imagery obtained by a sonar system is usually of lower quality. This can be attributed to the complexity of the underwater environment, such as strong reflection from seabeds, low cleanliness, inhomogeneity in the density of water. The strong reflection of seabeds makes the detection of objects that are close to the seabed very difficult. The inhomogeneity in the density of water can impair the transmission of the acoustic wave or even deviates the transmission from a straight path. The aim of this thesis is to design an advanced automatic system for the hunting of underwater mines.

1.1 Motivation

Due to the low quality of sonar imagery and the high varieties of different objects in the sonar imagery, the task of underwater target (in our case underwater mine) recognition has been undertaken by experienced human operators. With the expeditious development of autonomous underwater vehicles (AUVs) and the technological maturity of synthetic aperture sonar (SAS) systems [15] mounted on them, in the last two decades a huge volume of high quality sonar images have required processing. Therefore, the adoption of ATR in the underwater application is not only desirable but also indispensable, cf. [16–20]. An illustration of the ATR procedure is depicted in Fig. 1.1. In general, the ATR problem can be divided into two parts, namely detection and classification.

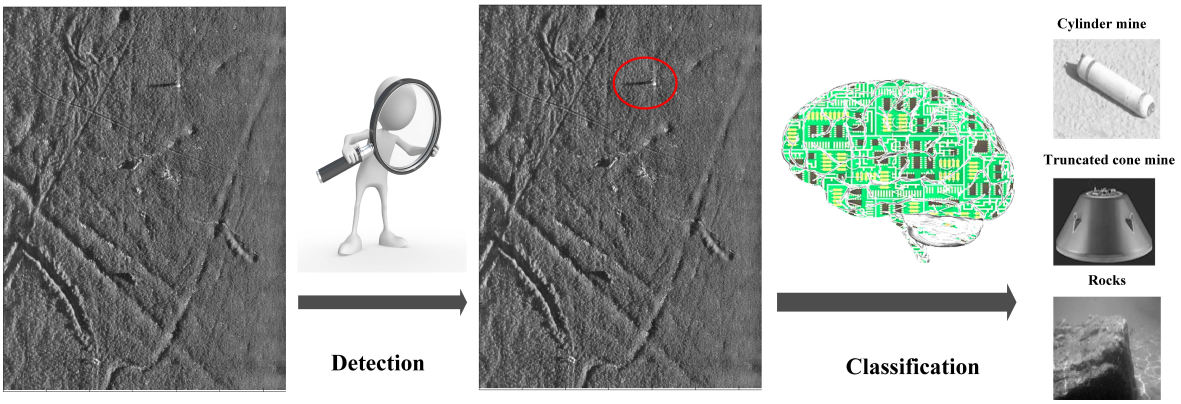


Figure 1.1. Automatic target recognition. From left to right: 1. The input data. 2. The detection of a target. 3. The target classification, i.e. whether the detected object is a mine or a rock.

Related works reported in the literature are mostly concentrated on traditional non-synthetic aperture sonar (NAS) systems [15]. Owing to the high cost of sea trials, the availability of real data has been constrained. Some authors even evaluated their approaches with the data collected from laboratory experiments. These kinds of experiments are usually carried out in a large water tank, e.g. [17]. Moreover, since the SAS systems are strategically related to military application, only a few authors in the SAS research field are willing to publish their studies. The well-known automatic detection and automatic classification (ADAC) system is adopted in this thesis. Among those published studies, most of them elaborate only the details of one or two nodes in the ADAC system.

Hence, we are motivated to present a complete overview of the ADAC system, and its application to the SAS data, which was collected by ATLAS Elektronik Bremen GmbH during several sea trials. The ADAC system is going to be described in detail as well as the contributions.

1.2 Introduction

A complete ADAC system contains four steps as shown in Fig. 1.2: mine-like object (MLO) detection, image segmentation, feature extraction and mine type classification. A range of techniques [20–23] has been developed for the purpose of target detection in the literature and they can be applied to the first step of MLO detection. If sufficient amounts of target examples are available, techniques such as supervised detection, template matching [20,21] and matched filters [22] can be applied. The success of template

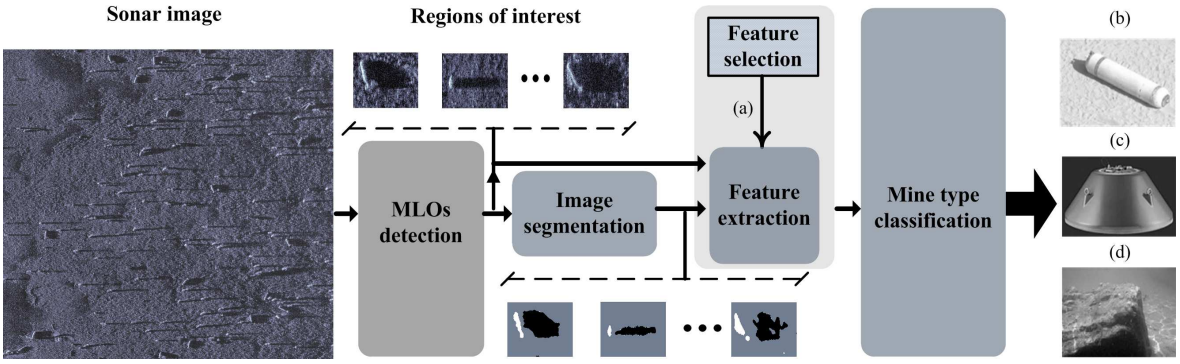


Figure 1.2. The illustration of the ADAC system. The contributions of this thesis are focused on image segmentation, feature selection and mine type classification. The feature selection is an indispensable step during the design of an ADAC system and (a) it controls the feature extraction step to extract useful features. The output of the system is the type of the MLO, i.e. (b) a cylinder mine, (c) a truncated cone mine or (d) a rock.

matching and matched filters depends on the similarity of the training data to the test data. Furthermore, Coiras *et al.* [23] proposed a supervised target detection by training on augmented reality data. The limited availability of real target samples is overcome by generating more samples that are created by augmented reality simulation [24]. After the MLO detection, those regions possibly containing MLOs are found, and they are called the regions of interest (ROI). The ROI are extracted and forwarded to the subsequent steps, i.e. the image segmentation and the feature extraction. Techniques like [18, 25] are employed in the step of image segmentation to segment the images of the ROI into highlights, shadows and backgrounds. The segmentation results are utilized for geometrical feature extraction. The goal of the feature extraction step is to prepare the inputs for the mine type classification step. In addition to the segmentation results, the images of the ROI are also taken into consideration for the extraction of texture features. A considerable amount of features have been proposed for the object recognition in the literature [26–32]. Due to the *curse of dimensionality* [33] shown in Fig. 1.3, the feature selection is necessary during the design of the ADAC system. Its result is used to guide the feature extraction so that only those useful features are extracted. With a number of appropriate features, the MLOs can be represented as points in the feature space in such a manner that the neighboring MLOs belong to the same classes and those of different classes are far away from each other. Finally, learning machines [34–36] are trained to classify those MLOs into different types, e.g. cylinder mines, truncated cone mines and rocks. For the sake of achieving a stable performance, an ensemble learning scheme is adopted. A number of learning machines are trained and the final classification result is obtained by combining the outputs of those trained learning machines.

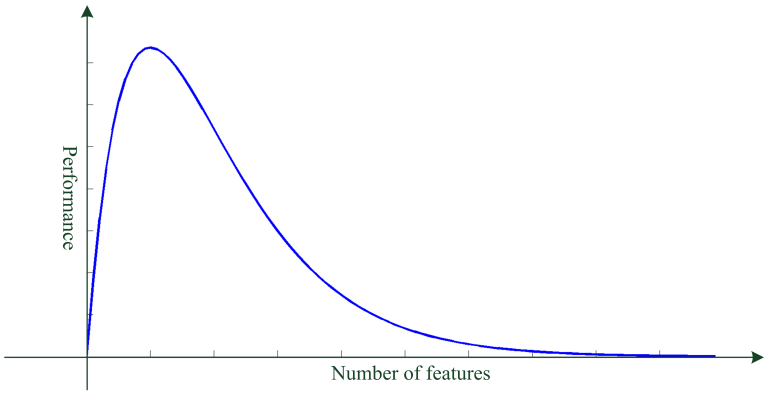


Figure 1.3. The curse of dimensionality. After the maximum point, the increase of feature number leads to a degradation of classification performance rather than improvement.

1.3 State of the Art

Our contributions to the design of a reliable ADAC system involves research in image segmentation, feature extraction, feature selection and mine type classification.

Numerous techniques have been developed for the purpose of image segmentation. Thresholding, e.g. [37–42], is a simple technique to divide images into different segments. Basically, a number of rigid thresholds should be set. The membership of pixels belonging to different classes depends on the comparison between the pixel intensities and the thresholds. Some authors have proposed to adapt the setting of thresholds to local characteristics. Due to the high level noise in the sonar imagery, the results are not satisfactory. The shape information of MLOs can be distorted. More complicated techniques such as [43–48] have attained success in the literature for a wide range of applications. They are able to provide satisfactory results with the data of high SNR, for instance the photos taken by digital cameras or satellite imagery. However, only a few publications, e.g. [6, 49], have referred to the application to SAS imagery.

The extraction of features has already been extensively discussed in the literature, cf. [26–32, 50–53]. Most of them are not specifically designed for the underwater targets. Among those for the underwater applications, many authors focused their feature extraction on the shadows. This is because the highlights are less discriminable than the shadows in the imagery acquired by NAS systems.

For feature selection, the methods such as dimensionality reduction [54,55] and feature subset selection [56–59] have been developed to reduce the dimensionality of feature

space. Firstly, the dimensionality reduction techniques, e.g. [54, 55], are vulnerable to the data scaling. Secondly, a method belonging to the category of feature subset selection requires evaluation metrics to assess the goodness of features. Mostly, either the classification accuracy obtained by a classifier (i.e. wrapper method) or a relevance measure (i.e. filter method) is utilized as the evaluation metrics. The wrapper methods can be computationally intensive and the associated selections are classifier dependent. As for the filter methods, many relevance measures have been proposed in the literature. However, many of them do not precisely evaluate the redundancy among features. Moreover, it is often the case that the most relevant features selected according to certain relevance measures do not necessarily always provide the best classification performance over various classifiers. Hence, it would be necessary to select a *suitable* classifier to match the features obtained according to a certain relevance measure. Unfortunately, this kind of correlation between relevance measures and classifiers is unknown.

As for the classifiers, researchers have kept on developing new learning machines, e.g. [60–62], or improving the existing learning algorithms, cf. [63,64]. Most of them claimed in their works that their proposals are superior to the others. However, the *No Free Lunch Theorem* [65] has already stated that there are no general optimal classifiers. Individual classifiers could attain the success to a certain degree in specific applications. Furthermore, it has also been observed that the sets of objects misclassified by different classifiers would not necessarily overlap. Hence, there are extensive studies dedicated to the topic of ensemble learning [66–85] in the last three decades.

1.4 Contributions

- **EM approach assisted by DST**: An approach called E-DS-M is developed for sonar imagery segmentation, in which an intermediate step (I-step) between the E- and M-steps of the expectation-maximization (EM) algorithm is introduced. In the I-step, a Dempster-Shafer theory based clustering is carried out so that the spatial correlation between neighboring pixels is considered. The likelihood function given by Sanjay-Gopal *et al.* [46] is employed and the Gaussian mixture is substituted by a generalized mixture model (Pearson system). As far as we know, it is the first time that the Pearson system is applied to SAS imagery for the image segmentation purpose. The adaption of Dempster-Shafer theory based clustering to the I-step is derived in detail and this approach provides us with reliable segmentation results with fewer EM iteration steps.

- **A summary of features used for underwater applications**: All of the features considered by us for the underwater object recognition have been reviewed and documented in this thesis. We have employed not only the geometrical features of the shadows but also of the highlights. In addition, a number of novel geometrical features are proposed. The correlation between highlights and shadows is also taken into account. The texture features of the ROI are also included in the feature set due to the fact that the deployment of objects on the seabed can change its texture characteristics.
- **Sophisticated filter method for feature selection**: We choose the mutual information (MI), the modified Relief weight (mRW) that is rooted in the Relief algorithm [86] and the Shannon information entropy to compose a new feature relevance measure, namely the composite relevance measure (CRM). Since the avoidance of *underfitting* and *overfitting* [87] is of great importance, the Shannon information entropy is adopted to control the complexity of feature selections. The CRM is capable of providing a comprehensive evaluation of the feature relevance.
- **Dempster-Shafer theory assisted ensemble learning in SAS imagery**: A reliable classifier combination scheme based on Dempster-Shafer theory is developed. Due to the fact that the training process of learning algorithms is not always optimal, the acquired classification results may contain uncertainty. This uncertainty can be elegantly modeled by *ignorance* in the framework of Dempster-Shafer theory. A basic belief assignment (BBA) is proposed to convert the outputs of classifiers to belief values.

1.5 Publications

The following publications have been produced during the period of PhD candidacy.

Internationally Refereed Journal Articles

- T. Fei, D. Kraus and A.M. Zoubir "Contributions to Automatic Target Recognition Systems For Underwater Mine Classification", *IEEE Transactions on Geoscience and Remote Sensing 2014*, Accepted.
- T. Fei and D. Kraus "Dempster-Shafer Theory Supported EM Approach For Sonar Image Segmentation", *Transactions on Systems, Signals & Devices* (SSN 1861-5252), Vol. 9, No. 3, pp.1-43, 2014.

Internationally Refereed Conference Papers

- T. Fei, A.F. Tchinda, B. Lehmann and D. Kraus, "On Sonar Image Processing Techniques for Anomaly Detection in Underwater Constructions", *the 8th European Conference on Synthetic Aperture Radar*, Aachen, Germany, Jun. 2010.
- T. Fei and D. Kraus, "An Expectation-Maximization Approach Assisted by Dempster Shafer Theory and its Application to Sonar Image Segmentation", *IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP 2012)*, Kyoto, Japan, Mar. 2012.
- T. Fei and D. Kraus, "An Evidence Theory Supported Expectation-Maximization Approach for Sonar Image Segmentation", *IEEE International Multi-Conference on Systems, Signals & Devices (SSD)*, Chemnitz, Germany, Mar. 2012.
- T. Fei, D. Kraus and P. Berkel "A New Idea On Feature Selection And Its Application To The Underwater Object Recognition", *the 11th European Conference on Underwater Acoustics*, Edinburgh, U.K., Jul. 2012.
- T. Fei, D. Kraus and A. M. Zoubir "A Novel Feature Selection Approach Applied To Underwater Object Classification", *European Signal Processing Conference*, Bucharest, Romania, Aug. 2012.
- T. Fei, D. Kraus and Abdelhak M. Zoubir "A Hybrid Relevance Measure for Feature Selection and Its Application to Underwater Objects Recognition", *IEEE International Conference on Image Processing (ICIP)*, Orlando, USA, Sep. 2012.
- T. Fei, D. Kraus and I. Aleksi "An Expectation-Maximization Approach Applied to Underwater Target Detection", *ICoURS'12 - International Conference on Underwater Remote Sensing*, Brest, France, Oct. 2012.
- T. Fei, D. Kraus and P. Berkel "A New Idea On Feature Selection And Its Application To The Underwater Object Recognition", *Proceedings of Meetings on Acoustics (POMA)*, Vol. 17, pp. 70071-70078, Jan. 2013.

1.6 Thesis Overview

The thesis outline is as follows. Chapter 2 describes the E-DS-M algorithm for sonar imagery segmentation. The generalized mixture model using the Pearson system is presented. After a brief introduction to the Dempster-Shafer theory, the derivation of

adapting Dempster-Shafer theory based clustering technique to the intermediate step between the E- and M-steps of expectation-maximization algorithm is detailed.

Chapter 3 provides a summary of the features used by us for the underwater object recognition. The extraction of features is explained and their characteristics are analyzed. In addition to those in the literature, we have proposed several geometrical features that are suitable to our application and their motivations are also elaborated.

In Chapter 4, a sophisticated filter method for feature selection is developed. The derivation and motivation of a composite relevance measure is comprehensively explained. In order to avoid the NP-hard problem during the search for optimal features, a heuristic scheme called sequential forward search is chosen for our filter method.

An ensemble learning with the assistance of Dempster-Shafer theory is presented in Chapter 5. We have novelly devised a basic belief assignment to convert the outputs of classifiers to belief values. All of the information acquired from different classifiers is fused by Dempster's rule.

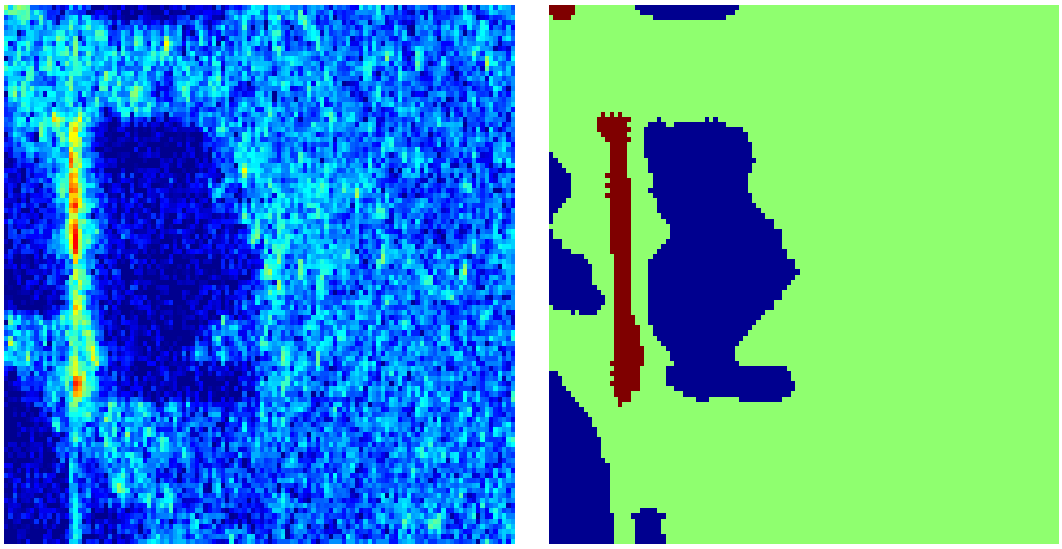
Conclusions are drawn in Chapter 6 and an outlook for future work is suggested as well.

Chapter 2

Sonar Imagery Segmentation

This chapter deals with MLOs detection as shown in Fig. 1.2. Following the MLO detection step, it is the second step along the process chain of the ADAC system. The accuracy of the segmentation in this step has a great influence on the performance of follow-on steps. Therefore, a reliable method is required in this step to extract the highlights and shadows which could be created by MLOs.

The image segmentation refers to the procedure of grouping image pixels into several classes. Those pixels belonging to the same homogeneous regions are assigned the same labels so that the sonar images will be divided into several regions, i.e. highlights, shadows and backgrounds. There is a segmentation example illustrated in Fig. 2.1. The



(a) (b)

Figure 2.1. An example of image segmentation. (a): SAS image containing a cylinder mine. (b): The segmentation result of the image on the left side. The labels for the background pixels are depicted in green, the shadow labels in blue and the highlight labels in red.

image contains a cylinder mine. The highlights, shadows and backgrounds are depicted in red, blue and green, respectively. Apparently, other than the largest shadow created by the cylinder mine, there are several clutters around the boundary of the image. They could be created by image noise or some natural objects (such as rocks).

In the literature, there are numerous segmentation techniques. Due to the high level noise in the sonar imagery, simple techniques, such as thresholding [37, 39, 39, 41] and k-means [88], might distort the shape information of MLOs, which is very important for mine type classification. Alternatively, the energy based active contour, e.g. [43,89], is another popular approach for image segmentation. However, according to our investigations, it is not optimal for the application in sonar images. Moreover, statistics based approaches [48, 90, 91] have employed maximum *a posteriori* probability estimation to fulfill the task of image segmentation. The posterior probability function usually contains two parts to describe the conditional probability of the image pixel intensities given the class labels of pixels and the spatial correlation between the labels of neighboring pixels. A Markov random field (MRF) approach is mostly involved [92] in the posterior probability function to cope with the spatial dependency between pixel labels through the implementation of a Gibbs distribution. The setting of parameters adopted in Gibbs distributions for controlling the relationship between neighboring pixels is still open. Usually, they are set according to the experience gathered from specific applications. Mignotte *et al.* in [48] have used a least squares technique to estimate the parameters. This estimation requires the histogramming of neighborhood configurations, which is a time-consuming process. Besides, the conditional probability of image pixel intensities is typically modeled by Gaussian, gamma and Weibull distributions, which are often not adequate to approximate the statistics of the data obtained from real measurements.

The EM algorithm [93] has been acting as a popular image segmentation approach for a long time, cf. [44,47]. In order to consider the spatial correlation between neighboring pixels, Zhang *et al.* [44] substitute the pixel class probability provided by the M-step of the previous iteration with an MRF based estimate. Later, Boccignone *et al.* [47] construct by inserting an anisotropic diffusion step [94] between each E- and M-step the so-called diffused expectation-maximization (DEM) scheme. With the assistance of the *a priori* knowledge that neighboring pixels are likely to be assigned with the same labels, neighboring pixels should have similar probabilities in the mixture distribution model. An anisotropic denoising filter is applied to probability levels so that the outliers with respect to their neighborhood are excluded, while the real edges of the image are still preserved. The application of such a denoising filter in DEM is not able to reliably exclude all of the noisy clusters in sonar images due to the fact that the variation of pixel intensities is high even for neighboring pixels. It is also possible to enlarge the object region because of the blur effect of denoising filters.

Most recently, the DST has been applied to the image segmentation [95–97]. In [95–97] the segmentation of color images is considered, which can be divided into image components of R, G and B. These three image components are used as information sources.

The belief structures in [95,96] are composed based on the assumption of Gaussian distribution. The mean and variance of the Gaussian distribution are estimated with the help of a simple thresholding technique [98] for each class. However, this estimation of the Gaussian distribution's parameters is not optimal for images with low signal-to-noise ratios. Besides, the fuzzy C-Mean algorithm is used for the segmentation of RGB images in [97]. The fuzzy membership is taken as basic belief assignment. Since the fuzzy membership can be interpreted rather as a particular plausibility function in the Dempster-Shafer evidence theory [99], it is improper to take the fuzzy membership as basic belief assignment.

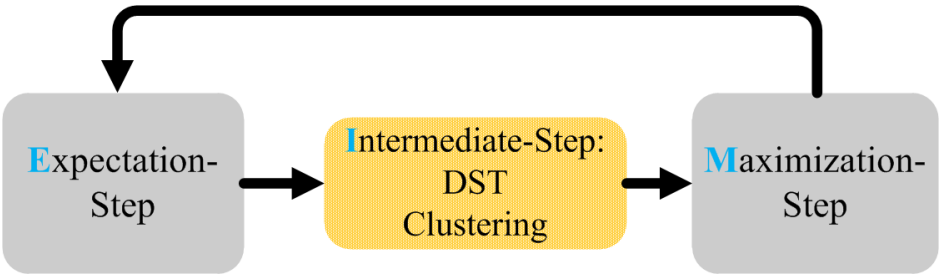


Figure 2.2. There is a generalized I-step inserted between the E- and M-step of the EM algorithm.

In this chapter, the macro-structure of DEM is employed and its diffusion step is generalized to an intermediate step (I-step) as presented in Fig. 2.2. The likelihood function of Sanjay-Gopal *et al.* is chosen. The correlation between pixels which are spatially far away from each other is decoupled. Furthermore, the classical Gaussian mixture is replaced by a generalized mixture model, whose components are chosen from a Pearson system [100]. There is a set of eight types of distribution in a Pearson system. The components of the mixture model are no longer required to be of the same distribution type. Therefore, the generalized mixture model is more flexible to approximate the statistics of sonar data. In addition, we apply the Dempster-Shafer theory based clustering technique in an I-step. The neighbors of a pixel are considered as pieces of evidence that support the hypotheses regarding the class label of this pixel.

This chapter is organized as follows. In Sec. 2.1 the image model is introduced. The maximum likelihood estimation, the Pearson system and EM algorithm are presented in Sec. 2.2. The spatial dependency among pixels is explained in Sec. 2.3. The proposed segmentation method using the Dempster-Shafer theory based clustering technique is given in Sec. 2.4. Finally, numerical studies are carried out using SAS images in Sec. 2.5. The results of our approach are compared to those in the literature. In order to make the analysis more convincing, a quantitative assessment is made with the assistance of the evaluation measure for image segmentation. Conclusions are drawn in Sec. 2.6.

2.1 Image Model

Since noise is inevitable in the real world, the image is corrupted by noise, and we call it observation. Let u_i be the intensity of the i-th pixel in the observation,

$$u_i = \mathfrak{u}_i + \epsilon_i, \tag{2.1}$$

where $\mathfrak{u}_i \in \mathfrak{U}$ denotes the intensity of pixel i in the unknown noise-free image, and ϵ_i is additive noise and $\mathfrak{U}$ is the set of all possible states of $\mathfrak{u}_i$. Let $\boldsymbol{\mathcal{L}}$ be a set of labels with $|\boldsymbol{\mathcal{L}}| = M_l$. Given the observation, our task of image segmentation is to assign to each $\mathfrak{u}_i$ a membership label $l_i \in \boldsymbol{\mathcal{L}}$. In our application, the $\boldsymbol{\mathcal{L}}$ contains three states which denote shadow, background and highlight, respectively $\boldsymbol{\mathcal{L}} = \{1, 2, 3\}$. Since the noise-free image is definitive and the noise added to pixels is uncorrelated, the observation, $\{u_i\}$, is conditionally independent given the $\{l_i\}$. The spatial correlation among pixels is reflected in the dependency among their labels.

2.2 Maximum Likelihood Estimation

For notational convenience, we denote noisy image/observation as a vector $\mathbf{u} = (u_1, ..., u_i, ..., u_{N_u})^T$, where N_u is the number of pixels in the image, $i \in \boldsymbol{\mathcal{I}} = \{1, 2, ..., N_u\}$. Analogously, the corresponding labels are represented by $\mathbf{l} = (l_1, ..., l_i, ..., l_{N_u})^T$. The conditional distribution of u_i given l_i is

$$p(u_i|l_i = j) = f_U(u_i|\boldsymbol{\psi}_j), \tag{2.2}$$

where $j \in \boldsymbol{\mathcal{L}}$, f_U is an arbitrary probability density function, and $\boldsymbol{\psi}_j$ is the parameter required for the distribution when $l_i = j$. An indicator vector $\mathbf{r}_i = (r_{i,1}, ...r_{i,j}, ..., r_{i,M_l})^T \in \{\mathbf{e}_1, ..., \mathbf{e}_{M_l}\}$ for $M_l = |\boldsymbol{\mathcal{L}}|$ is defined, and we have the probability

$$\begin{aligned} p(l_i = j) &= p(\mathbf{r}_i = \mathbf{e}_j), & (2.3)\\ &= \pi_{i,j}, & (2.4) \end{aligned}$$

where $\pi_{i,j}$ is a mixing coefficient with $0 \leq \pi_{i,j} \leq 1, \sum_{j=1}^{M_l} \pi_{i,j} = 1$ and $\mathbf{e}_j$ is a unit vector whose j-th component is 1. Then, Equation (2.2) can be written as

$$p(u_i|r_{i,j} = 1) = f_U(u_i|\boldsymbol{\psi}_j), \tag{2.5}$$

which can also be formalized in the form as follows:

$$p(u_i|\mathbf{r}_i) = \prod_{j=1}^{M_l} f_U(u_i|\boldsymbol{\psi}_j)^{r_{i,j}}. \tag{2.6}$$

The joint distribution of $\mathbf{r}_i$ and u_i is given by $p(\mathbf{r}_i)p(u_i|\mathbf{r}_i)$ with

$$p(\mathbf{r}_i) = \prod_{j=1}^{M_l} \pi_{i,j}^{r_{i,j}}, \tag{2.7}$$

and the marginal distribution of u_i is obtained by summing the joint distribution over all the possible states of $\mathbf{r}_i$,

$$\begin{aligned} p(u_i) &= \sum_{j=1}^{M_l} p(\mathbf{r}_i = \mathbf{e}_j)p(u_i|\mathbf{r}_i = \mathbf{e}_j) \\ &= \sum_{j=1}^{M_l} \pi_{i,j} f_U(u_i|\boldsymbol{\psi}_j). \end{aligned} \tag{2.8}$$

The distribution of u_i is presented by Equation (2.8), and it is usually called distribution mixture model. In this thesis, we allow the f_U to be chosen from a Pearson system $\boldsymbol{\mathcal{F}} = \{F_1, ..., F_8\}$. The choice of the distribution type out of $\boldsymbol{\mathcal{F}}$ is going to be detailed in the next subsection.

2.2.1 Pearson System

Let U be a real random variable whose distribution can be modeled by a Pearson system. The probability density function $f(u)$ satisfying the differential equation [100],

$$\frac{1}{f}\frac{df}{du} = -\frac{a+u}{\mathfrak{a}_0 + \mathfrak{a}_1 u + \mathfrak{a}_2 u^2}, \tag{2.9}$$

belongs subject to the setting of the parameters $a, \mathfrak{a}_0, \mathfrak{a}_1$ and $\mathfrak{a}_2$ to one of the eight possible distribution types of a Pearson system. The solutions of Equation 2.9 depend on the roots of the characteristic equation

$$\mathfrak{a}_0 + \mathfrak{a}_1 u + \mathfrak{a}_2 u^2 = 0. \tag{2.10}$$

The details about Equation (2.9) are stated as follows.

1. The *Type I* distribution (F_1) corresponds to the case that both roots of Equation (2.10) are real, and of opposite signs, i.e. $\frac{\mathfrak{a}_1^2 - 4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2} > 0$ and $\frac{\mathfrak{a}_0}{\mathfrak{a}_2} < 0$ for $\mathfrak{a}_2 \neq 0$. The density function can be given as

$$f(u) = \begin{cases} \frac{1}{B(\tau_1,\tau_2)} \frac{(u-b_1)^{\tau_1-1}(b_2-u)^{\tau_2-1}}{(b_2-b_1)^{\tau_1+\tau_2-1}}, & \text{for } u \in [b_1, b_2] \\ 0, & \text{otherwise} \end{cases}, \tag{2.11}$$

with

$$\begin{aligned} b_1 &= -\frac{\mathfrak{a}_1}{2\mathfrak{a}_2} - 0.5\sqrt{\frac{\mathfrak{a}_1^2 - 4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2}}, \\ b_2 &= -\frac{\mathfrak{a}_1}{2\mathfrak{a}_2} + 0.5\sqrt{\frac{\mathfrak{a}_1^2 - 4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2}}, \\ \tau_1 &= \frac{a + b_1}{\mathfrak{a}_2(b_2 - b_1)} + 1, \\ \tau_2 &= -\frac{a + b_2}{\mathfrak{a}_2(b_2 - b_1)} + 1, \end{aligned}$$

and $B(\tau_1, \tau_2)$ is beta function. This distribution is also called *Beta distribution of the first kind*.

2. The *Type II* distribution (F_2) is a particular case of F_1 with $\tau_1 = \tau_2$, and the density function is as follows:

$$f(u) = \begin{cases} \frac{1}{B(\tau,\tau)} \frac{(u-b_1)^{\tau-1}(b_2-u)^{\tau-1}}{(b_2-b_1)^{2\tau-1}}, & \text{for } u \in [b_1, b_2] \\ 0, & \text{otherwise} \end{cases}, \tag{2.12}$$

with

$$\begin{aligned} \tau &= \frac{a + b_1}{\mathfrak{a}_2(b_2 - b_1)} + 1, \\ b_1 &= -a - 0.5\sqrt{\frac{\mathfrak{a}_1^2 - 4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2}}, \\ b_2 &= -a + 0.5\sqrt{\frac{\mathfrak{a}_1^2 - 4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2}}. \end{aligned}$$

3. The *Type III* distribution (F_3) corresponds to the case $\mathfrak{a}_2 = 0$ (and $\mathfrak{a}_1 \neq 0$). In this case, the density function is

$$f(u) = \begin{cases} \frac{1}{\tau_1\Gamma(\tau_2)} \left(\frac{u-\tau_3}{\tau_1}\right)^{\tau_2-1} e^{-(u-\tau_3)/\tau_1}, & \text{for } u \geq \tau_3 \\ 0, & \text{otherwise} \end{cases}, \tag{2.13}$$

with

$$\begin{aligned} \tau_1 &= \mathfrak{a}_1, \\ \tau_2 &= \frac{1}{\mathfrak{a}_1}\left(\frac{\mathfrak{a}_0}{\mathfrak{a}_1} - a\right) + 1, \\ \tau_3 &= -\frac{\mathfrak{a}_0}{\mathfrak{a}_1} \end{aligned}$$

and Γ denotes the gamma function. This distribution is also termed as *gamma distribution*.

4. The *Type IV* distribution (F_4) refers to the case in which Equation (2.10) does not have real roots, i.e. $\mathfrak{a}_1^2 - 4\mathfrak{a}_0\mathfrak{a}_2 < 0$.

$$f(u) = \mathfrak{N}_1 \left(\tau_1 + \mathfrak{a}_2(u+\tau_2)^2\right)^{-(1/2\mathfrak{a}_2)} \exp\left(-\frac{a-\tau_2}{\sqrt{\tau_1\mathfrak{a}_2}} \arctan\left(\sqrt{\frac{\mathfrak{a}_2}{\tau_1}}(u+\tau_2)\right)\right), \tag{2.14}$$

with the factor $\mathfrak{N}_1$ such that $\int_{\mathcal{R}} f(u)du = 1$ and

$$\begin{aligned} \tau_1 &= \mathfrak{a}_0 - \frac{\mathfrak{a}_1^2}{4\mathfrak{a}_2}, \\ \tau_2 &= \frac{\mathfrak{a}_1}{2\mathfrak{a}_2}. \end{aligned}$$

Unfortunately, there is no common statistical distribution whose density function has a form as the one in Equation (2.14). Woodward proposed a simple mathematical form to approximate this distribution [101] as follows:

$$f_{\text{app}}(u) = \tilde{\mathfrak{N}}_1 \left(1 + \frac{u^2 - \tilde{\lambda}}{\tilde{a}^2}\right)^{-\tau_3} \exp\left(-\tau_4 \arctan\left(\frac{u-\tilde{\lambda}}{\tilde{a}}\right)\right), \tag{2.15}$$

where

$$\begin{aligned} \tau_3 &= \frac{1}{2\mathfrak{a}_2}, \\ \tilde{b} &= 2(\tau_3 - 1), \\ \tau_4 &= \frac{2\mathfrak{a}_1(1-\tau_3)}{\sqrt{4\mathfrak{a}_0\mathfrak{a}_2 - \mathfrak{a}_1^2}}, \\ \tilde{a} &= \sqrt{\frac{\tilde{b}^2(\tilde{b}-1)}{\tilde{b}^2 + \tau_4^2}}, \\ \tilde{\lambda} &= \frac{\tilde{a}\tau_4}{\tilde{b}} \end{aligned}$$

with the factor $\tilde{\mathfrak{N}}_1$ such that $\int_{\mathcal{R}} f_{\text{app}}(u)du = 1$ [102],

$$\tilde{\mathfrak{N}}_1 = \frac{\Gamma(\tau_3)}{\sqrt{\pi}\tilde{a}\Gamma(\tau_3 - 0.5)} \left\| \frac{\Gamma(\tau_3 + i\tau_4/2)}{\Gamma(\tau_3)} \right\|^2, \tag{2.16}$$

where the i in this equation denotes the imaginary unit.

5. The *Type V* distribution (F_5) corresponds to the case where $\mathfrak{a}_1^2 = 4\mathfrak{a}_0\mathfrak{a}_2$. The associated distribution density function is

$$f(u) = \begin{cases} \frac{\tau_1}{\Gamma(\tau_2)} \left(\tau_1 \left(u + \frac{\mathfrak{a}_1}{2\mathfrak{a}_2}\right)\right)^{-\tau_2 - 1} \exp\left(\frac{-2}{\tau_1\left(u + \frac{\mathfrak{a}_1}{2\mathfrak{a}_2}\right)}\right), & \text{for } u \geq -\frac{\mathfrak{a}_1}{2\mathfrak{a}_2} \\ 0, & \text{otherwise} \end{cases}, \tag{2.17}$$

with

$$\begin{aligned}\tau_1 &= \frac{\mathfrak{a}_2}{a-\frac{\mathfrak{a}_1}{2\mathfrak{a}_2}},\\ \tau_2 &= \frac{1}{\mathfrak{a}_2}-1.\end{aligned}$$

This distribution is also termed as *inverse gamma distribution*.

6. The *Type VI* distribution (F_6) corresponds to the case in which the roots of Equation (2.10) are real and of the same sign, i.e. $\frac{\mathfrak{a}_1^2-4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2}\geq 0$ and $\frac{\mathfrak{a}_0}{\mathfrak{a}_2}>0$. The associated distribution density function is

$$f(u)=\begin{cases}\frac{\tau_4^{\tau_2}}{B(\tau_1,\tau_2)}\frac{(u-\tau_3)^{\tau_1-1}}{(u-(\tau_3-\tau_4))^{\tau_1+\tau_2}}, & \text{for } u\geq\tau_3\\ 0, & \text{otherwise}\end{cases}, \tag{2.18}$$

with

$$\begin{aligned}\tau_1 &= -\frac{a-\frac{1}{2\mathfrak{a}_2}\left(\mathfrak{a}_1-\sqrt{\mathfrak{a}_1^2-4\mathfrak{a}_0\mathfrak{a}_2}\right)}{\sqrt{\mathfrak{a}_1^2-4\mathfrak{a}_0\mathfrak{a}_2}}+1,\\ \tau_2 &= \frac{1}{\mathfrak{a}_2}-1\\ \tau_3 &= -\frac{1}{2\mathfrak{a}_2}\left(\mathfrak{a}_1-\sqrt{\mathfrak{a}_1^2-4\mathfrak{a}_0\mathfrak{a}_2}\right),\\ \tau_4 &= \sqrt{\frac{\mathfrak{a}_1^2-4\mathfrak{a}_0\mathfrak{a}_2}{\mathfrak{a}_2^2}} \quad \text{for } \mathfrak{a}_2\neq 0.\end{aligned}$$

This distribution is also called *Beta distribution of the second kind*.

7. The *Type VII* distribution (F_7) is the case in which $\mathfrak{a}_1=a=0,\mathfrak{a}_0>0$, and $\mathfrak{a}_2>0$. The corresponding density function is given as [103]

$$f(u)=\mathfrak{N}_2\left(\frac{\tau_2^2}{2\tau_1}\left(1+\left(\frac{u}{\tau_2}\right)^2\right)\right)^{-\tau_1}, \tag{2.19}$$

where

$$\begin{aligned}\tau_1 &= \frac{1}{2\mathfrak{a}_2},\\ \tau_2 &= \sqrt{2\tau_1\mathfrak{a}_0},\\ \mathfrak{N}_2 &= \frac{1}{\mathfrak{B}(0.5,\tau_1-0.5)}\frac{\tau_2^{2\tau_1-1}}{(2\tau_1)^{\tau_1}},\end{aligned}$$

with $\mathfrak{B}(\tau_3,\tau_4)=\int_0^\infty\frac{u^{\tau_3-1}}{(1+u)^{(\tau_3+\tau_4)}}du$.

8. The *Type VIII* distribution (F_8) is the case where $\mathfrak{a}_1 = \mathfrak{a}_2 = 0$. Thus the associated density function is

$$f(u) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(u-\mu)^2}{2\sigma^2}}, \tag{2.20}$$

with $\mu = -a$ and $\sigma^2 = \mathfrak{a}_0$. Obviously, it is the *Gaussian distribution.*

As summarized above, the determination of the distribution type is dependent on the values of the parameters $a, \mathfrak{a}_0, \mathfrak{a}_1$ and $\mathfrak{a}_2$. However, they are usually unknown *a priori.* Johnson *et al.* demonstrated that it is possible to express $a, \mathfrak{a}_0, \mathfrak{a}_1$ and $\mathfrak{a}_2$ in terms of central moments as follows [45]:

$$a = \frac{(\mathfrak{s}_2 + 3)\sqrt{\mathfrak{s}_1\zeta_2}}{10\mathfrak{s}_2 - 12\mathfrak{s}_1 - 18} - \mu, \tag{2.21}$$

$$\mathfrak{a}_0 = \frac{\zeta_2(4\mathfrak{s}_2 - 3\mathfrak{s}_1) - \mu(\mathfrak{s}_2 + 3)\sqrt{\mathfrak{s}_1\zeta_2} + \mu^2(2\mathfrak{s}_2 - 3\mathfrak{s}_1 - 6)}{10\mathfrak{s}_2 - 12\mathfrak{s}_1 - 18}, \tag{2.22}$$

$$\mathfrak{a}_1 = \frac{(\mathfrak{s}_2 + 3)\sqrt{\mathfrak{s}_1\zeta_2} - 2\mu(2\mathfrak{s}_2 - 3\mathfrak{s}_1 - 6)}{10\mathfrak{s}_2 - 12\mathfrak{s}_1 - 18}, \tag{2.23}$$

$$\mathfrak{a}_2 = \frac{2\mathfrak{s}_2 - 3\mathfrak{s}_1 - 6}{10\mathfrak{s}_2 - 12\mathfrak{s}_1 - 18}, \tag{2.24}$$

where μ and ζ_n are given by

$$\mu = E[U], \tag{2.25}$$

$$\zeta_n = E[(U-\mu)^n], \text{for } n = 2, 3, \text{ and } 4, \tag{2.26}$$

and the $\mathfrak{s}_1$ and $\mathfrak{s}_2$ are defined as

$$\mathfrak{s}_1 = \frac{(\zeta_3)^2}{(\zeta_2)^3}, \tag{2.27}$$

$$\mathfrak{s}_2 = \frac{\zeta_4}{(\zeta_2)^2}. \tag{2.28}$$

Hence, the classification of the distribution type, which was based on the setting of $a, \mathfrak{a}_0, \mathfrak{a}_1$ and $\mathfrak{a}_2$, can be done via the moments. The advantage of this conversion is that in practical applications the central moments can be estimated from the data. Based on the moments, the rule can be reformulated as follows,

$$\begin{cases} f \in F_1, & \text{for } \lambda < 0, \\ f \in F_2, & \text{for } \mathfrak{s}_1 = 0 \text{ and } \mathfrak{s}_2 < 3, \\ f \in F_3, & \text{for } 2\mathfrak{s}_2 - 3\mathfrak{s}_1 - 6 = 0, \\ f \in F_4, & \text{for } 0 < \lambda < 1, \\ f \in F_5, & \text{for } \lambda = 1, \\ f \in F_6, & \text{for } \lambda > 1, \\ f \in F_7, & \text{for } \mathfrak{s}_1 = 0 \text{ and } \mathfrak{s}_2 > 3, \\ f \in F_8, & \text{for } \mathfrak{s}_1 = 0 \text{ and } \mathfrak{s}_2 = 3, \end{cases} \tag{2.29}$$

where λ is defined as

$$\lambda = \frac{\mathfrak{s}_1(\mathfrak{s}_2+3)^2}{4(4\mathfrak{s}_2-3\mathfrak{s}_1)(2\mathfrak{s}_2-3\mathfrak{s}_1)(2\mathfrak{s}_2-3\mathfrak{s}_1-6)}. \tag{2.30}$$

2.2.2 Expectation-Maximization Algorithm

In this subsection, the observation is considered as statistically independent, the joint conditional density of the observations can be formed as

$$p(\mathbf{u}|\mathbf{\Psi}) = \prod_{i=1}^{N_u}\sum_{j=1}^{M_l}\pi_{i,j}f_U(u_i|\boldsymbol{\psi}_j), \tag{2.31}$$

where $\mathbf{\Psi} = \left(\boldsymbol{\psi}_1^T, ..., \boldsymbol{\psi}_{M_l}^T\right)^T$. The EM algorithm [93] is a powerful method to maximize the likelihood in Equation (2.31). It requires the specification of complete data $\mathfrak{z} = \left(\mathbf{u}^T, \mathbf{r}_1^T, ..., \mathbf{r}_{N_u}^T\right)^T$ in contrast to the incomplete data/observation $\mathbf{u}$. Moreover, we define the parameter vectors $\boldsymbol{\pi}_i = (\pi_{i,1}, ..., \pi_{i,M_l})^T$, $\mathbf{\Pi} = \left(\boldsymbol{\pi}_1^T, ..., \boldsymbol{\pi}_{N_u}^T\right)^T$ and $\mathbf{\Phi} = \left(\mathbf{\Pi}^T, \mathbf{\Psi}^T\right)^T$. In deriving an EM algorithm, the conditional density function for the complete data $\mathfrak{z}$ is required. With the help of Equation (2.6) and Equation (2.7), we have the conditional pdf of complete data

$$p(\mathfrak{z}|\mathbf{\Phi}) = \prod_{i=1}^{N_u}\prod_{j=1}^{M_l}\left[\pi_{i,j}f_U(u_i|\boldsymbol{\psi}_j)\right]^{r_{i,j}}, \tag{2.32}$$

where f_U belongs to some type of distribution out of the set $\boldsymbol{\mathcal{F}}$. The EM algorithm iterates itself between an E-step where a conditional expectation is computed and an M-step where the estimates of parameters (i.e. $\mathbf{\Pi}$ and $\mathbf{\Psi}$) are updated by maximizing this conditional expectation. The E-step is defined as

$$\begin{aligned} Q\left(\mathbf{\Phi}|\mathbf{\Phi}^{(k)}\right) &= E\left[\ln\left(p(\mathfrak{z}|\mathbf{\Phi})\right)|\mathbf{\Phi}^{(k)}, \mathbf{U}=\mathbf{u}\right], &(2.33)\\ &= E\left[\sum_{i=1}^{N_u}\sum_{j=1}^{M_l} r_{i,j}\left(\ln\pi_{i,j} + \ln f_U(u_i|\boldsymbol{\psi}_j)\right)|\mathbf{\Phi}^{(k)}, \mathbf{U}=\mathbf{u}\right], &(2.34)\end{aligned}$$

where $\mathbf{U} = (U_1, ..., U_{N_u})^T$ and $\mathbf{\Phi}^{(k)}$ denotes the parameters obtained in the k-th iteration. In order to compute the expectation in Equation (2.34), the distribution of $r_{i,j}$ is of interest to us. Similar as derived in Appendix C of [46], we have

$$\begin{aligned} p\left(r_{i,j}=1|\mathbf{\Phi}^{(k)}\right) &= E\left[r_{i,j}|\mathbf{\Phi}^{(k)}\right], &(2.35)\\ &= \frac{\pi_{i,j}^{(k)} f_U\left(u_i|\boldsymbol{\psi}_j^{(k)}\right)}{\sum_{m=1}^{M_l}\pi_{i,m}^{(k)} f_U\left(u_i|\boldsymbol{\psi}_m^{(k)}\right)}. &(2.36)\end{aligned}$$

For notation convenience, let $w_{i,j}^{(k)} = p\left(r_{i,j} = 1|\mathbf{\Phi}^{(k)}\right)$. In the M-step, the $\mathbf{\Phi}^{(k)}$ should be updated with

$$\mathbf{\Phi}^{(k+1)} = \arg\max_{\mathbf{\Phi}} Q\left(\mathbf{\Phi}|\mathbf{\Phi}^{(k)}\right). \tag{2.37}$$

For pixel i, conditioned on $\sum_{j=1}^{M_l} \pi_{i,j} = 1$, a Lagrange multiplier $\mathfrak{A}$ is introduced

$$\Lambda = \sum_{j=1}^{M_l} w_{i,j}^{(k)} \left[\ln \pi_{i,j} + f_U(u_i|\boldsymbol{\psi}^{(k)})\right] + \mathfrak{A}\left(\sum_{j=1}^{M_l} \pi_{i,j} - 1\right). \tag{2.38}$$

Through $\frac{\partial \Lambda}{\partial \pi_{i,j}} = 0$ for $j = 1, .., M_l$ we get

$$\frac{w_{i,j}^{(k)}}{\pi_{i,j}} + \mathfrak{A} = 0, \qquad j = 1, ..., M_l \tag{2.39}$$

$$\frac{\sum_{j=1}^{M_l} w_{i,j}^{(k)}}{\mathfrak{A}} = -1. \tag{2.40}$$

Solving Equations (2.39) and (2.40), we get the update of $\pi_{i,j}$,

$$\begin{aligned} \pi_{i,j}^{(k+1)} &= w_{i,j}^{(k)}, \\ &= \frac{\pi_{i,j}^{(k)} f_U\left(u_i|\boldsymbol{\psi}_j^{(k)}\right)}{\sum_{m=1}^{M_l} \pi_{i,m}^{(k)} f_U\left(u_i|\boldsymbol{\psi}_m^{(k)}\right)}, \end{aligned} \tag{2.41}$$

and the mean and the central moments are updated as follows,

$$\begin{aligned} \mu_j^{(k+1)} &= \frac{\sum_{i=1}^{N_u} u_i \pi_{i,j}^{(k+1)}}{\sum_{i=1}^{N_u} \pi_{i,j}^{(k+1)}}, \\ &= \frac{\sum_{i=1}^{N_u} u_i w_{i,j}^{(k)}}{\sum_{i=1}^{N_u} w_{i,j}^{(k)}}, \end{aligned} \tag{2.42}$$

$$\begin{aligned} \zeta_{n,j}^{(k+1)} &= \frac{\sum_{i=1}^{N} \left(u_i - \mu_j^{(k+1)}\right)^n \pi_{i,j}^{(k+1)}}{\sum_{i=1}^{N} \pi_{i,j}^{(k+1)}} \quad \text{for } n = 2, 3 \text{ and } 4, \\ &= \frac{\sum_{i=1}^{N} \left(u_i - \mu_j^{(k+1)}\right)^n w_{i,j}^{(k)}}{\sum_{i=1}^{N} w_{i,j}^{(k)}}, \end{aligned} \tag{2.43}$$

where μ_j and $\zeta_{n,j}$ are the mean value and the n-th central moment of the pixels belonging to class j, respectively. With the results in Equations (2.42) and (2.43), $\mathfrak{s}_1$, $\mathfrak{s}_2$ and λ can be obtained. Accordingly, the distribution types of f_U and their associated parameters can be determined as described in Sec. 2.2.1 for the next EM iteration.

There are two examples of segmentation results with mixture models shown in Fig. 2.3. The segmentation result obtained by Gaussian mixture is presented in subfigure (c) and

the one corresponding to generalized mixture is in subfigure (d). Compared with the generalized mixture model, Gaussian mixture provides a segmentation result whose background region is more heavily eroded by clutters. Moreover, the pdf estimates illustrated in subfigure (b) demonstrate that the generalized mixture can better approximate the statistics of an SAS image.

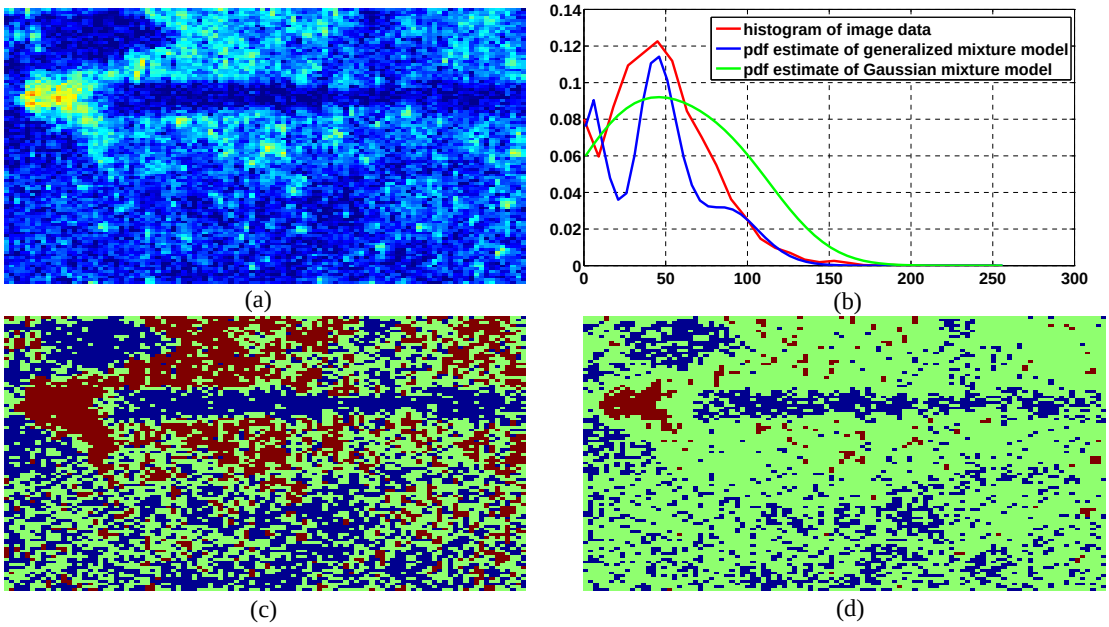


Figure 2.3. An example to illustrate the comparison between segmentation results obtained by the EM with generalized mixture model and the EM with Gaussian mixture model. (a): A sonar image containing a truncated cone mine. (b): The pdf estimates obtained by the EM with Gaussian mixture model and the EM with generalized mixture model. (c): The segmentation result obtained by the EM with Gaussian mixture model. (d): The segmentation result obtained by the EM with generalized mixture model.

However, it is obvious that both of the segmentation results shown in Fig. 2.3 are not satisfactory. They are "dirty". In a segmentation result, the object region (i.e. highlight or shadow) should be smooth and connected. Ideally, there should be as few pixels as possible in the background region which are classified as highlights or shadows due to the image noise. In order to fulfill this requirement, the correlation between neighboring pixels should be considered.

2.3 Spatial Dependency among Pixels

For the sake of "clean" segmentation results, the spatial correlation among pixels has to be taken into account in this section. The labeling of pixel i is influenced by the

states of its neighbors. The Markov random field has been widely employed to model this relation. Most recently, the Dempster-Shafer theory is also applied to remove the clutters in the segmentation results. We assume that a pixel depends on its neighbors in such a manner that *the neighboring pixels with similar intensities are likely to have identical labels* or *a pixel is probably to be assigned to the group which contains the majority of it neighbors*. In view of this manner, the clustering techniques relying on Markov random field and Dempster-Shafer theory are derived to model the spatial correlation among pixels in the following two subsections.

2.3.1 Markov Random Field

Let $\boldsymbol{\mathcal{N}}_i$ be the neighborhood of pixel i such that for its j-th neighbor $\eta_{i,j}$ we have $\eta_{i,j} \in \boldsymbol{\mathcal{N}}_i$ and $i \in \boldsymbol{\mathcal{N}}_{\eta_{i,j}}$. This pair of $\{i, \eta_{i,j}\}$ is known as a clique [104]. In this thesis, the second order neighborhood is employed as shown in Fig. 2.4. On the left side, the second order neighborhood containing eight neighbors is illustrated. On the right side, the eight associated cliques within the neighborhood $\boldsymbol{\mathcal{N}}_i$ are presented. In most cases, the cliques are classified into four different types as depicted on the right side of Fig. 2.4.

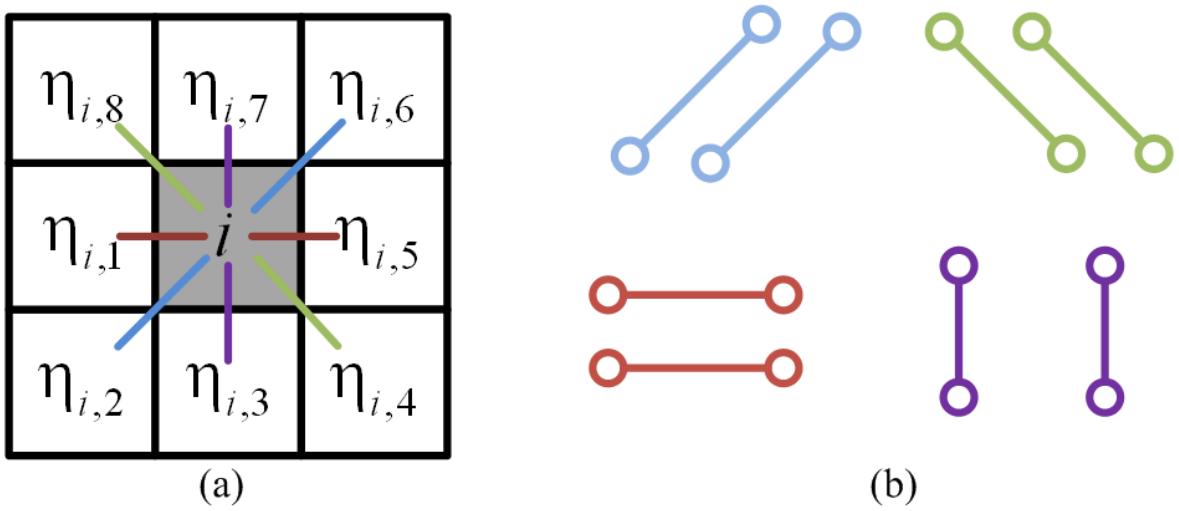


Figure 2.4. The second order neighborhood of pixel i, $\boldsymbol{\mathcal{N}}_i$, and the associated cliques. (a): The second order neighborhood, $\eta_{i,1}, ..., \eta_{i,8} \in \boldsymbol{\mathcal{N}}_i$. (c): The four kinds of cliques. From left to right and from top to bottom, their relationships are specified by $\beta_1, \beta_2, \beta_3$ and β_4, respectively.

The Hammersley-Clifford theorem [105] reveals that there is a one-to-one correspondence between MRF and Gibbs random field, which is defined by the Gibbs distribution.

Hence, *a priori* probability p_l is able to be conveniently modeled as follows:

$$p_l(l_i) = \frac{1}{Z} e^{-\Upsilon(l_i)}, \tag{2.44}$$

where Z is a normalization constant and $\Upsilon(l_i)$ is an energy function

$$\Upsilon(l_i) = \mathfrak{S}(l_i, \mathfrak{L}_i)^T \boldsymbol{\beta}, \tag{2.45}$$

where $\mathfrak{L}_i = \left(l_{\eta_{i,1}}, ..., l_{\eta_{i,8}}\right)^T$ denotes the configuration of neighborhood $\boldsymbol{\mathcal{N}}_i$, $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ and $\mathfrak{S}(l_i, \mathfrak{L}_i)$ is given by

$$\begin{aligned} \mathfrak{S}(l_i, \mathfrak{L}_i) \quad = \quad & \big(\mathfrak{S}(l_i, l_{\eta_{i,2}}) + \mathfrak{S}(l_i, l_{\eta_{i,6}}), \mathfrak{S}(l_i, l_{\eta_{i,4}}) + \mathfrak{S}(l_i, l_{\eta_{i,8}}), \\ & \mathfrak{S}(l_i, l_{\eta_{i,1}}) + \mathfrak{S}(l_i, l_{\eta_{i,5}}), \mathfrak{S}(l_i, l_{\eta_{i,3}}) + \mathfrak{S}(l_i, l_{\eta_{i,7}})\big)^T, \\ \text{with} \quad & l_{\eta_{i,1}}, ..., l_{\eta_{i,8}} \in \boldsymbol{\mathcal{N}}_i, \end{aligned} \tag{2.46}$$

where $\mathfrak{S} = 1 - \delta_{\text{Kronecker}}$, and $\delta_{\text{Kronecker}}$ is the Kronecker delta function. Then, the spatial correlation can be determined by the Gibbs distribution in Equation (2.44). It is usually chosen by a MAP estimator as the prior in posterior probability density function, which is detailed in the following.

In the Bayesian theorem [106], one can combine the prior information with the likelihood to obtain a posterior probability,

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}, \tag{2.47}$$

where in our application the conditional probability $p(\mathbf{u}|\mathbf{l})$ is the likelihood and the spatial dependency specified in terms of Gibbs distribution $p_l(\mathbf{l})$ is the prior. Then for a given observation (i.e. a sonar image), we have

$$p(\mathbf{l}|\mathbf{u}) = \frac{p_l(\mathbf{l}) p(\mathbf{u}|\mathbf{l})}{p(\mathbf{u})}, \tag{2.48}$$

where $p(\mathbf{u}|\mathbf{l}) = \prod_{i=1}^{N_u} p(u_i|l_i)$, $p(\mathbf{l}) = \prod_{i=i}^{N_u} p_l(l_i)$ and the *Evidence* is a normalization factor to ensure that the total probability is 1. Hence, it is usually expressed as follows:

$$p(\mathbf{l}|\mathbf{u}) \propto \exp\left(-\mathcal{E}(\mathbf{u}, \mathbf{l}, \boldsymbol{\beta})\right), \tag{2.49}$$

where $\mathcal{E}(\mathbf{u}, \mathbf{l}, \boldsymbol{\beta})$ is the posterior energy. There is an isotropic model [91] in which the $\mathcal{E}(\mathbf{u}, \mathbf{l}, \boldsymbol{\beta})$ has the same β for all cliques in the neighborhood, i.e.

$$\mathcal{E}_{\text{isotropic}}(\mathbf{u}, \mathbf{l}, \beta) = -\sum_{i=1}^{N_u} \ln p(u_i|l_i) + \sum_{i=1}^{N_u} \sum_{\eta \in \boldsymbol{\mathcal{N}}_i} \beta \left(1 - \delta_{\text{Kronecker}}(l_i - l_\eta)\right), \tag{2.50}$$

where the β has to be set *a priori* based on empirical knowledge. Moreover, Reed *et al.* [18] proposed an anisotropic model,

$$\mathcal{E}_{\text{anisotropic}}(\mathbf{u}, \mathbf{l}, \boldsymbol{\beta}) = -\sum_{i=1}^{N_u} \ln p(u_i|l_i) + \sum_{i=1}^{N_u} \mathfrak{S}(l_i, \mathfrak{L}_i)^T \boldsymbol{\beta}, \tag{2.51}$$

where $\beta_1, ...\beta_4$ could be different. The last two terms of their energy function is omitted since there is no prior knowledge available in our application about the object orientation and object size. For a given neighborhood configuration $\mathfrak{L}$, the ratio of the probabilities of pixel i being labeled with j and j' can be calculated as

$$\ln \frac{p(l_i = j|\mathfrak{L})}{p(l_i = j'|\mathfrak{L})} = \left(\mathfrak{S}(l_i = j, \mathfrak{L}) - \mathfrak{S}(l_i = j', \mathfrak{L})\right)^T \boldsymbol{\beta}. \tag{2.52}$$

For each possible neighborhood configuration, the term on the left side of Equation (2.52) can be approximated by using a simple histogramming as follows:

$$\frac{p(l_i = j|\mathfrak{L})}{p(l_i = j'|\mathfrak{L})} = \frac{\#\{i' \in \boldsymbol{\mathcal{I}} : l_{i'} = j, \mathfrak{L}_{i'} = \mathfrak{L}\}}{\#\{i' \in \boldsymbol{\mathcal{I}} : l_{i'} = j', \mathfrak{L}_{i'} = \mathfrak{L}\}}, \tag{2.53}$$

where # denotes the number of elements in the set. This creates an over-determined set of equations for the four unknowns, i.e. $\beta_1, \beta_2, \beta_3$ and β_4. It can be solved by a least squares technique.

2.3.2 Dempster-Shafer Theory Based Clustering

2.3.2.1 Basics about Dempster-Shafer Theory

In 1967, Arthur P. Dempster proposed a new concept of upper and lower probabilities [107]. His work remained hidden in the statistics literature until Glenn Shafer, one of Dempster's students, brought the material to a wider audience in his doctoral dissertation [108]. Although it has been more than forty years since then, the Dempster-Shafer theory is still not as familiar as the fuzzy logic to most engineers. Hence, it is worth providing some basics about the Dempster-Shafer theory before going into the details about our modeling of ensemble learning. The Dempster-Shafer theory (DST) is a mathematical theory of evidence. It allows one to combine the information from different pieces of evidence and arrive at a degree of belief which takes into account all the available evidence. In DST, the set containing all the hypotheses is called *the frame of discernment*. In this chapter, the pixels can be labeled by the elements out of the set $\boldsymbol{\mathcal{L}} = \{1, 2, 3\}$. Therefore, the set $\boldsymbol{\mathcal{L}}$ is the frame of discernment. The function

$\mathfrak{b} : 2^{\boldsymbol{\mathcal{L}}} \rightarrow [0,1]$ describing this belief portion assignment and satisfying the following conditions:

$$\mathfrak{b}(\emptyset) = 0, \tag{2.54}$$

$$\sum_{\boldsymbol{\Delta} \subseteq \boldsymbol{\mathcal{L}}} \mathfrak{b}(\boldsymbol{\Delta}) = 1, \tag{2.55}$$

is called basic belief assignment (BBA). The quantity $\mathfrak{b}(\boldsymbol{\Delta})$ can be understood as a measure for the belief portion assigned to the hypothesis that the correct answer is in $\boldsymbol{\Delta}$. However, no further information about the distribution of this amount of belief portion to the subsets of $\boldsymbol{\Delta}$ can be inferred. In other words, the $\mathfrak{b}(\boldsymbol{\Delta})$ does not make any additional claim about the hypothesis that the correct answer lies in a subset of $\boldsymbol{\Delta}$. Every $\boldsymbol{\Delta} \in 2^{\boldsymbol{\mathcal{L}}}$ that satisfies $\mathfrak{b}(\boldsymbol{\Delta}) > 0$ is called a *focal* element of the BBA. Based on the BBA, the belief function is defined by

$$Bel(\boldsymbol{\Delta}) = \sum_{\boldsymbol{\Delta}' \subseteq \boldsymbol{\Delta}} \mathfrak{b}(\boldsymbol{\Delta}'). \tag{2.56}$$

The quantity $Bel(\boldsymbol{\Delta})$ represents the total belief committed to the hypothesis $\boldsymbol{\Delta}$. It can easily be verified [63] that the $Bel(\boldsymbol{\Delta})$ and the $Bel(\bar{\boldsymbol{\Delta}})$ with $\bar{\boldsymbol{\Delta}} = \boldsymbol{\mathcal{L}} \backslash \boldsymbol{\Delta}$ do not necessarily add up to 1. It is a major difference from probability theory. Moreover, another quantity $Pl(\boldsymbol{\Delta}) = 1 - Bel(\bar{\boldsymbol{\Delta}})$ called plausibility is defined to describe the extent to which one fails to doubt in $\boldsymbol{\Delta}$,

$$Pl(\boldsymbol{\Delta}) = \sum_{\boldsymbol{\Delta}' \bigcap \boldsymbol{\Delta} \neq 0} \mathfrak{b}(\boldsymbol{\Delta}'). \tag{2.57}$$

Hence, the probability of hypothesis $\boldsymbol{\Delta}$ is bounded by Bel and Pl, $Bel(\boldsymbol{\Delta}) \leq P(\boldsymbol{\Delta}) \leq Pl(\boldsymbol{\Delta}), \forall \boldsymbol{\Delta} \subseteq \boldsymbol{\mathcal{L}}$.

Dempster's rule is a mathematical operation used to combine two BBAs induced by different pieces of evidence, $\mathfrak{b}_1$ and $\mathfrak{b}_2$,

$$\mathfrak{b}_{1 \oplus 2}(\boldsymbol{\Delta}) = \frac{\sum_{\boldsymbol{\Delta}_1 \bigcap \boldsymbol{\Delta}_2 = \boldsymbol{\Delta}} \mathfrak{b}_1(\boldsymbol{\Delta}_1) \mathfrak{b}_1(\boldsymbol{\Delta}_2)}{1 - \sum_{\boldsymbol{\Delta}_1 \bigcap \boldsymbol{\Delta}_2 = \emptyset} \mathfrak{b}_1(\boldsymbol{\Delta}_1) \mathfrak{b}_1(\boldsymbol{\Delta}_2)}, \tag{2.58}$$

where $\boldsymbol{\Delta}, \boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2 \in 2^{\boldsymbol{\mathcal{L}}}$. Since Dempster's rule is commutative and associative, the BBAs of diverse evidence can be combined sequentially in any arrangement. The decision-making of DST is still open. There exists an interval of probabilities bounded by Bel and Pl. Consequently, simple hypotheses can no longer be ranked according to their probabilities. Over the last thirty years, many proposals have been made to conquer this uncertainty on probabilities. In this chapter, we use the well-known pignistic probability [109] proposed by P. Smets, which has been verified by P. Smets

and R. Kennes in [110] as a convenient and justified mechanism for converting a BBA into a probability,

$$BetP(\mathbf{\Delta}) = \sum_{\mathbf{\Delta}' \subseteq \boldsymbol{\mathcal{L}}} \mathfrak{b}(\mathbf{\Delta}') \frac{|\mathbf{\Delta} \bigcap \mathbf{\Delta}'|}{|\mathbf{\Delta}'|}. \tag{2.59}$$

If the readers are interested in DST, more information can be found in [111].

2.3.2.2 Dempster-Shafer Theory Based Clustering

In the framework of DST, we model the neighbors as pieces of evidence. They provide support to the hypotheses that the pixel of interest (e.g. pixel i in the case given in Fig. 2.5) belongs to the same classes of these neighbors. As depicted in Figure 2.5, it is

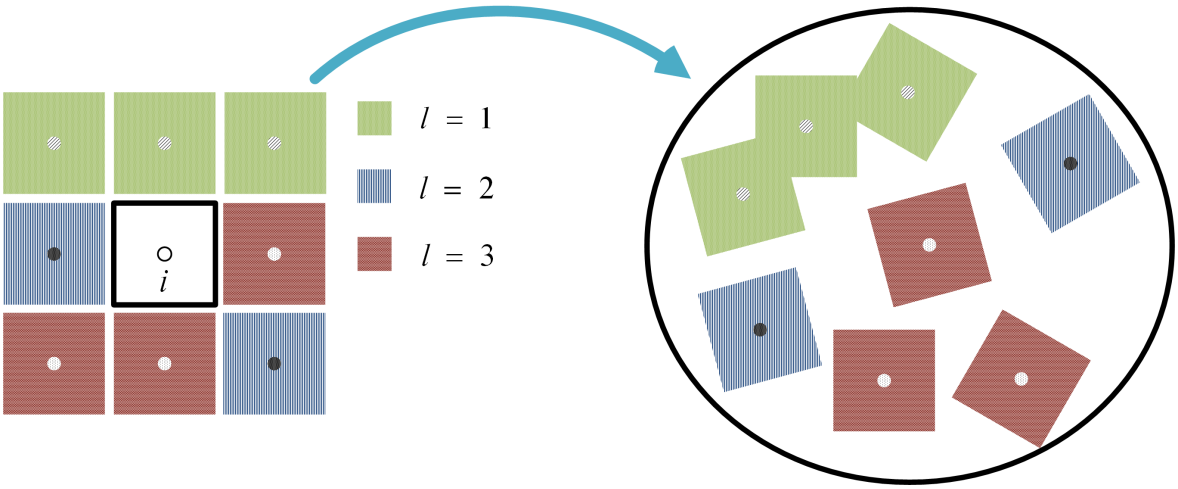


Figure 2.5. (a): The neighborhood configuration of pixel i, $\mathfrak{L}_i$. (b): The evidence pool.

a second order neighborhood of the pixel of interest, i.e. pixel i. All of its neighbors are labeled, and can be used as evidence. The amount of support provided by a neighbor η to the hypothesis that pixel i is assigned with the same label as pixel η relies on the difference between u_i and the average of all the $u_{i'}$ with $l_{i'} = l_\eta$. Hence, the variation caused by the noise contained in the observation of the neighbors can be minimized. Obviously, a small difference in the pixel intensities should indicate a great amount of support.

We model the support provided by the neighbors as follows. If a neighbor $\eta \in \boldsymbol{\mathcal{N}}_i$

belongs to class $l_\eta \in \boldsymbol{\mathcal{L}}$, its BBA is given as

$$\mathfrak{b}(\boldsymbol{\Delta}) = \begin{cases} \vartheta_\eta v_\eta, & \text{if} \quad \boldsymbol{\Delta} = \{l_\eta\}, \\ 1 - \vartheta_\eta v_\eta, & \text{if} \quad \boldsymbol{\Delta} = \boldsymbol{\mathcal{L}}, \\ 0, & \text{otherwise}, \end{cases} \tag{2.60}$$

where the ϑ_η and v_η are determined by

$$\vartheta_\eta = \frac{\exp(-\gamma_1 |u_\eta - \nu_i|)}{\max_{\eta' \in \boldsymbol{\mathcal{N}}_i} \exp(-\gamma_1 |u_{\eta'} - \nu_i|)}, \tag{2.61}$$

$$v_\eta = \exp\left(-\gamma_2 \frac{|u_i - \mu_{l_\eta}|}{\sigma_{l_\eta}}\right), \tag{2.62}$$

where μ_{l_η} and σ_{l_η} are the mean value and standard deviation of class l_η, ν_i is the median of the pixel intensity of $\boldsymbol{\mathcal{N}}_i$, and γ_1, γ_2 are positive constants. The v_η denotes the total belief portion which is able to be provided by the pixel $\eta \in \boldsymbol{\mathcal{N}}_i$, and the ϑ_η evaluates the quality of the evidence. This quality evaluation is based on the assumption that the information supplied by an outlier should be less plausible. The manner of how ϑ_η and v_η react to the parameters γ_1 and γ_2 is qualitatively presented in Fig. 2.6. The parameter γ_1 in Equation (2.61) manipulates the tolerance against outliers. If it

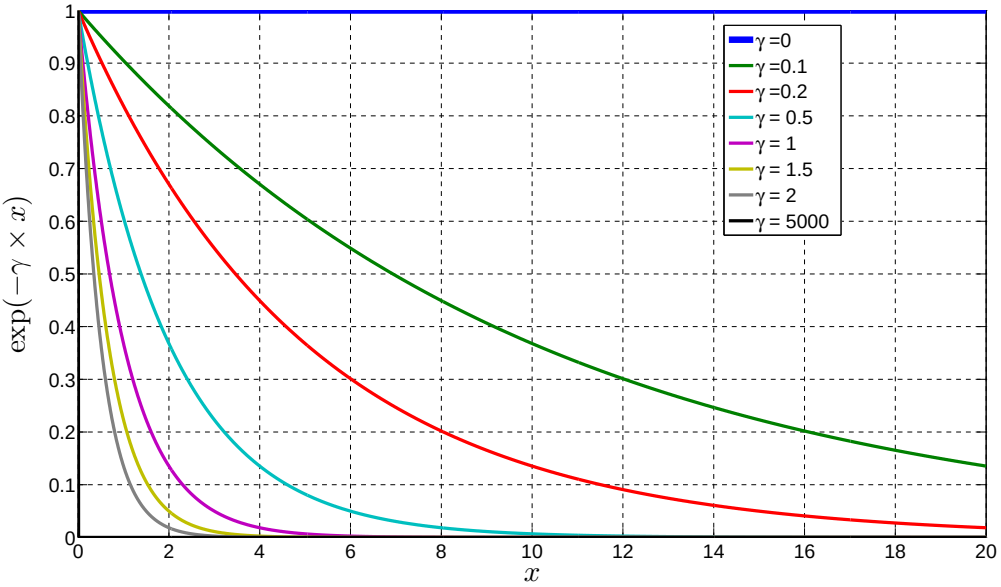


Figure 2.6. The illustration of function $\exp(-\gamma \times x)$.

approaches infinity, only those pixels whose intensity is identical to the median are taken into account. In contrast, when γ_1 equals zero, all the pixels are considered to be of the same quality. The parameter γ_2 in Equation (2.62) controls the total belief portion assignment. When it is increased, the assignment is more sensitive to the

distance between u_i and μ_{l_η}. The choice of γ_1, γ_2 will be justified in Sec. 2.5. Moreover, since v_η is distance dependent, it is necessary to normalize all the distance measures into the same scale by dividing the measures by σ_{l_η} as in Equation (2.62).

Dempster's rule considers all the possible combinations of elements out of the power set $2^{\boldsymbol{\mathcal{L}}}$. When the number of elements in the set $\boldsymbol{\mathcal{L}}$ increases, the time consumed for the BBA combination grows exponentially. There is an effective combination scheme for the simple BBA derived from Dempster's rule by Denoeux *et al.* in [63]. It considers only those elements which are focals of the combining BBAs:

$$\mathfrak{b}_{\text{total}}(\{l\}) = \frac{\mathfrak{b}^{(l)}(\{l\}) \prod_{l' \neq l} \mathfrak{b}^{(l')}(\boldsymbol{\mathcal{L}})}{\mathfrak{K}}, \tag{2.63}$$

$$\mathfrak{b}_{\text{total}}(\boldsymbol{\mathcal{L}}) = \frac{\prod_{l \in \boldsymbol{\mathcal{L}}} \mathfrak{b}^{(l)}(\boldsymbol{\mathcal{L}})}{\mathfrak{K}}, \tag{2.64}$$

where $\mathfrak{b}^{(l)}$ is given by

$$\mathfrak{b}^{(l)}(\{l\}) = 1 - \prod_{\eta \in \boldsymbol{\mathcal{N}}_i^l} \left(1 - \mathfrak{b}_\eta(\{l\})\right), \tag{2.65}$$

$$\mathfrak{b}^{(l)}(\boldsymbol{\mathcal{L}}) = \prod_{\eta \in \boldsymbol{\mathcal{N}}_i^l} \left(1 - \mathfrak{b}_\eta(\{l\})\right), \tag{2.66}$$

where $\boldsymbol{\mathcal{N}}_i^l \subseteq \boldsymbol{\mathcal{N}}_i$ is the set of neighbors in $\boldsymbol{\mathcal{N}}_i$ belonging to the class $l \in \boldsymbol{\mathcal{L}}$, $\mathfrak{b}_\eta$ is the BBA associated with the neighbor η and $\mathfrak{K}$ is the normalizing factor:

$$\mathfrak{K} = \sum_{l \in \boldsymbol{\mathcal{L}}} \prod_{l' \neq l} \mathfrak{b}^{(l')}(\boldsymbol{\mathcal{L}}) + (1 - |\boldsymbol{\mathcal{L}}|) \prod_{l \in \boldsymbol{\mathcal{L}}} \mathfrak{b}^{(l)}(\boldsymbol{\mathcal{L}}). \tag{2.67}$$

After the information combination, there should be a final decision made on the combined BBA $\mathfrak{b}_{\text{total}}$. We choose the most well known pignistic probability [109] for the sake of decision-making. Due to the fact that focals of $\mathfrak{b}_{\text{total}}$ are either elements of $\boldsymbol{\mathcal{L}}$ or $\boldsymbol{\mathcal{L}}$ itself, the results obtained from the pignistic level are identical to those from the BBA function. Thus, the decision-making for pixel i is given by

$$l_i = \arg\max_{l \in \boldsymbol{\mathcal{L}}} \mathfrak{b}_{\text{total},i}(\{l\}), \tag{2.68}$$

where $\mathfrak{b}_{\text{total},i}$ is the combined BBA associated with pixel i.

2.4 EM Algorithm Assisted with Dempster-Shafer Theory Based Clustering

In the previous sections, the generalized mixture model that assumes the independence among pixels, EM algorithm and the Dempster-Shafer theory based clustering are

presented. They have to be combined in the way illustrated in Figure 2.2. The idea is that before the output of E-step ($\{w_{i,j}|1 \leq i \leq N_u, j \in \boldsymbol{\mathcal{L}}\}$) is forwarded to M-step, it should be processed by the I-step (Dempster-Shafer theory based clustering) to incorporate the neighborhood information. The input of the M-step is substituted by $\{\bar{w}_{i,j}|1 \leq i \leq N_u, j \in \boldsymbol{\mathcal{L}}\}$,

$$\bar{w}_{i,j} = \begin{cases} 1, & l_i = j, \\ 0, & l_i \neq j. \end{cases} \tag{2.69}$$

For unsupervised methods, the initialization is of great importance. Since the gamma mixture has been widely adopted in the processing of radar [112] and sonar imagery [25, 49] to approximate the statistics of non-negative data, we initialize the model with gamma mixture. Hence, the proposed method called E-DS-M can be summarized as follows,

Step 1. The gamma mixture model is chosen for the initialization and its parameters are estimated as in [113]

Step 2. Run E-step with the help of Equation (2.35), and obtain $\left\{w_{i,j}^{(k)}\right\}$

Step 3. Perform a hard decision on $\left\{w_{i,j}^{(k)}\right\}$, then get $\left\{l_i^{(k)}\right\}$

Step 4. Determine the BBA as shown in Equations (2.60), (2.61) and (2.62)

Step 5. Combine the BBAs with the assistance of Equations (2.63), (2.64), (2.65), (2.66) and (2.67)

Step 6. Determine the l_i and $\left\{\bar{w}_{i,j}^{(k)}\right\}$ by Equation (2.68) and Equation (2.69), respectively

Step 7. Forward the $\left\{\bar{w}_{i,j}^{(k)}\right\}$ to the M-step, substitute the $w_{i,j}^{(k)}$ with $\bar{w}_{i,j}^{(k)}$ in Equations (2.42) and (2.43), and estimate the central moments of each class, $\mu_j^{(k+1)}$ and $\zeta_{n,j}^{(k+1)}$ with $n = 2, 3$ and 4 using Equations (2.42) and (2.43)

Step 8. Determine the types of f_U in Equation (2.32) with the help of Equations (2.27), (2.28), (2.30) and (2.29)

Step 9. Go back to **Step 2** until the results converge or the number of maximum iteration steps is reached

The comparison of EM and E-DS-M is represented in Figure 2.7. In subfigure (b), the estimated pdfs are illustrated. It is apparent that the inclusion of spatial correlation among pixels does not increase the accuracy of the pdf estimation. However, it improves the segmentation results by removing most of the clutters in the background region.

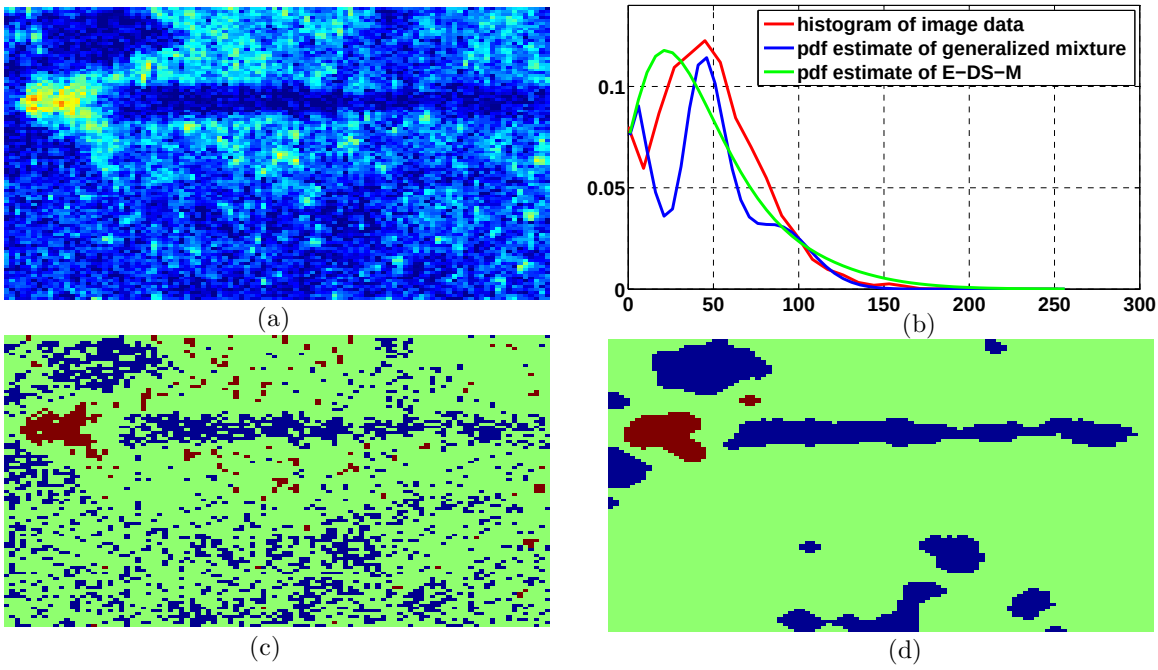


Figure 2.7. The comparison between segmentation results obtained by the EM with generalized mixture model and the E-DS-M. (a): The SAS image containing a truncated cone mine. (b): The pdf estimates obtained by the EM with generalized mixture model and the E-DS-M. (c): The segmentation result provided by the EM with generalized mixture model. (d): The segmentation result provided by the E-DS-M.

2.5 The Numerical Studies of E-DS-M

Numerical tests are carried out on both real SAS data and synthetic data. The ripple-like sediment is a great challenge for sonar image segmentation. Owing to the high cost of sea trials, the availability of real sonar data is limited. We have only the SAS data that is obtained from sea trails launched on flat sediments. Thus, we simulate the SAS data with ripple-like sediment to verify the reliability of E-DS-M. It is found in our study that E-DS-M can provide almost perfect results on ripple-like sediments. The performance gain against the methods in the literature can be easily observed. Therefore, there is no necessity to use additional measures for the evaluation of the results obtained from synthetic data. In contrast, due to the complexity of real SAS images, a quantitative measure dedicated to image segmentation is required for the performance evaluation.

We choose the MAP estimator which adopts an isotropic model for neighborhood (MAP-ISO) given by Equation (2.50), the MAP estimator proposed by Reed *et al.*

(MAP-Reed) using the energy function in Equation (2.51) and DEM [47] for comparison. The maximization problem of the two posterior probabilities of MAP-ISO and MAP-Reed is solved by the ICM algorithm.

2.5.1 Evaluation Measure for Image Segmentation

We employ in this chapter the variation of information (VI) [114] to evaluate the segmentation results.

Let $\mathcal{S}$ denote a segmentation of the image, and it divides the $\boldsymbol{\mathcal{D}} = \{u_1, u_2, ..., u_{N_u}\}$ into groups $\boldsymbol{\mathcal{S}}_1, \boldsymbol{\mathcal{S}}_2, ..., \boldsymbol{\mathcal{S}}_{M_l}$ such that

$$\boldsymbol{\mathcal{S}}_j \bigcap \boldsymbol{\mathcal{S}}_k = \emptyset \qquad \text{and} \qquad \bigcup_{j=1}^{M_l} \boldsymbol{\mathcal{S}}_j = \boldsymbol{\mathcal{D}}, \tag{2.70}$$

where $j \neq k$. The number of pixels can also be given as $N_u = |\boldsymbol{\mathcal{D}}|$ and the pixel number in $\boldsymbol{\mathcal{S}}_j$ is $N_{u,j} = |\boldsymbol{\mathcal{S}}_j|$. Let another segmentation be $\mathcal{S}'$ and it segments the image into $\boldsymbol{\mathcal{S}'}_1, \boldsymbol{\mathcal{S}'}_2, ..., \boldsymbol{\mathcal{S}'}_{j'}, ..., \boldsymbol{\mathcal{S}'}_{M'_l}$ with group size of $N'_{u,j'}$. The number of pixels in the intersection of $\boldsymbol{\mathcal{S}}_j$ and $\boldsymbol{\mathcal{S}}'_{j'}$ is denoted as $N_{u,jj'}$,

$$N_{u,jj'} = \left\| \boldsymbol{\mathcal{S}}_j \bigcap \boldsymbol{\mathcal{S}}'_{j'} \right\|. \tag{2.71}$$

VI measures the difference between two segmentations in terms of the information entropy,

$$I_{\mathrm{VI}}(\mathcal{S}, \mathcal{S}') = H(\mathcal{S}) + H(\mathcal{S}') - 2I(\mathcal{S}, \mathcal{S}'), \tag{2.72}$$

where $H(\mathcal{S})$ and $I(\mathcal{S}, \mathcal{S}')$ are defined as

$$H(\mathcal{S}) = -\sum_{j=1}^{M_l} \frac{N_{u,j}}{N_u} \log_2 \frac{N_{u,j}}{N_u}, \tag{2.73}$$

$$I(\mathcal{S}, \mathcal{S}') = \sum_{j=1}^{M_l} \sum_{j'=1}^{M'_l} \frac{N_{u,jj'}}{N_u} \log_2 \frac{\frac{N_{u,jj'}}{N_u}}{\frac{N_{u,j}}{N_u} \frac{N_{u,j'}}{N_u}}. \tag{2.74}$$

It is shown in Figure 2.8 that the VI provides us the measure on dissimilarity between two segmentations $\mathcal{S}$ and $\mathcal{S}'$. If they are identical, the entropies $H(\mathcal{S})$ and $H(\mathcal{S}')$ will totally overlap with each other. The mutual information $I(\mathcal{S}, \mathcal{S}')$ equals to $H(\mathcal{S})$. In this case, $I_{\mathrm{VI}}(\mathcal{S}, \mathcal{S}') = 0$. We substitute the result of $\mathcal{S}$ with the ground truth. Let $\mathcal{S}'$ denote the segmentation result obtained by different segmentation methods. Consequently, if the segmentation method works ideally, we have the evaluation measure $I_{\mathrm{VI}} = 0$.

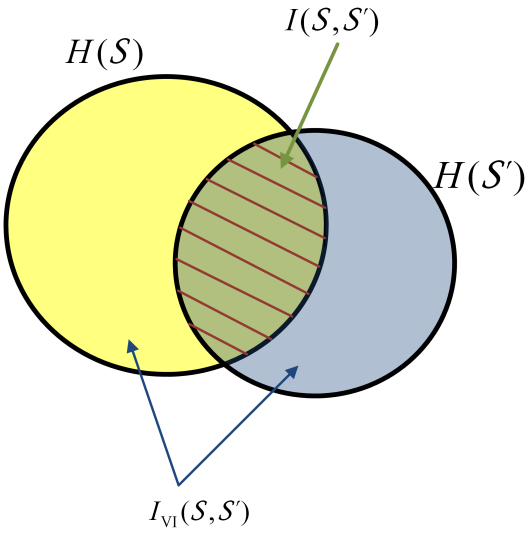


Figure 2.8. The illustration of VI.

There are examples of different segmentation results in Fig. 2.9. The first one on the top left of this figure is the ground truth. The VI of the following 15 segmentation results are computed against this ground truth, and their values are depicted in Fig. 2.10. The segmentation result 5 in Fig. 2.9 is identical to the ground truth. Thus, its VI is 0.

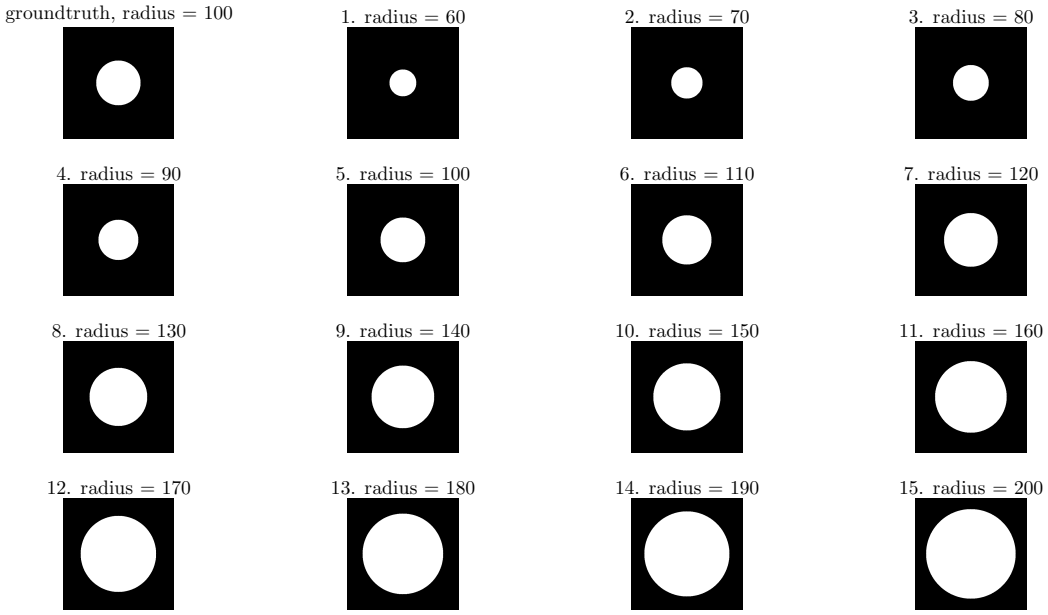


Figure 2.9. An example of the comparison among different segmentation results. The one on the top left of the figure is ground truth. We calculate the VI of the following segmentation results against this ground truth.

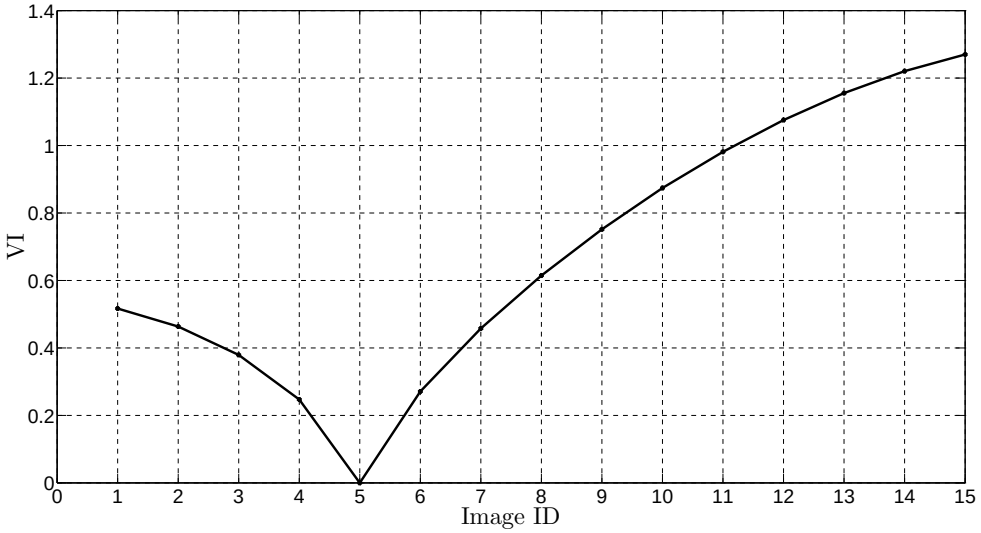


Figure 2.10. The VI associated with the segmentation results in Fig. 2.9.

2.5.2 Experiments on Real SAS Images

There are eight real SAS images containing MLOs presented in Fig. 2.11. Their corresponding ground truths are given in Fig. 2.12. Their dimensions are 100×100 pixels.

In order to visualize the impact of γ_1 and γ_2, we vary them to reveal how the E-DS-M reacts to the tuning of parameters. We compute the I_{VI} of all the test images in Fig. 2.11 and present the averages of I_{VI} over the eight images in Fig. 2.13.

Obviously, although the variation of γ_2 in (2.62) has some influence on the performance of image segmentation, it is neither significant nor definite. In contrast, the performance of image segmentation is highly dependent on the setting of γ_1 in (2.61). As γ_1 grows, more neighbors are recognized as outliers and their support to the corresponding hypotheses is suppressed. The consequence is that the useful information embodied in the neighbors could be ignored and the segmentation results of the E-DS-M are impaired. There is significant performance degradation around $\gamma_1 = 0.2$. According to the results in Fig. 2.13, the E-DS-M has a satisfying performance when γ_1 is around 0.1. We find that the optimal parameter setting in this test is $\gamma_1 = 0.1$ and $\gamma_2 = 1.4$.

An example to illustrate the impact of γ_1 is shown in Fig. 2.14. It is an example of Image 7. For simplicity, the parameter γ_2 is set to 1. It can be observed that the increasing of γ_1 introduces much clutter in the background.

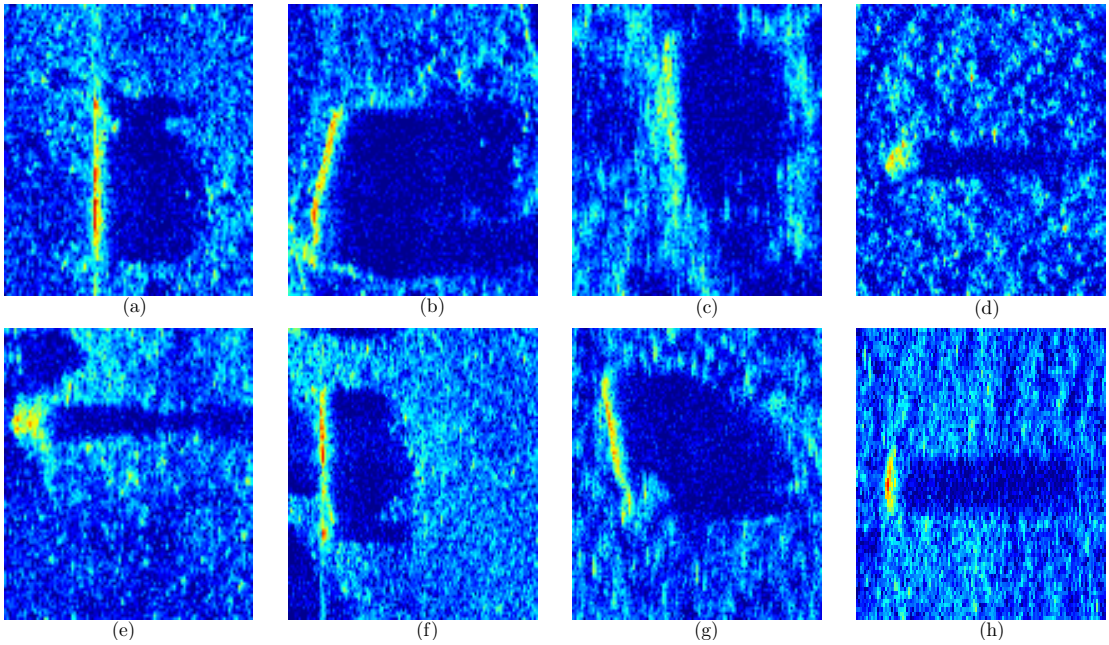


Figure 2.11. The SAS images used for the evaluation of image segmentation methods. Subfigures (a)−(b) denote test image 1 to test image 8.

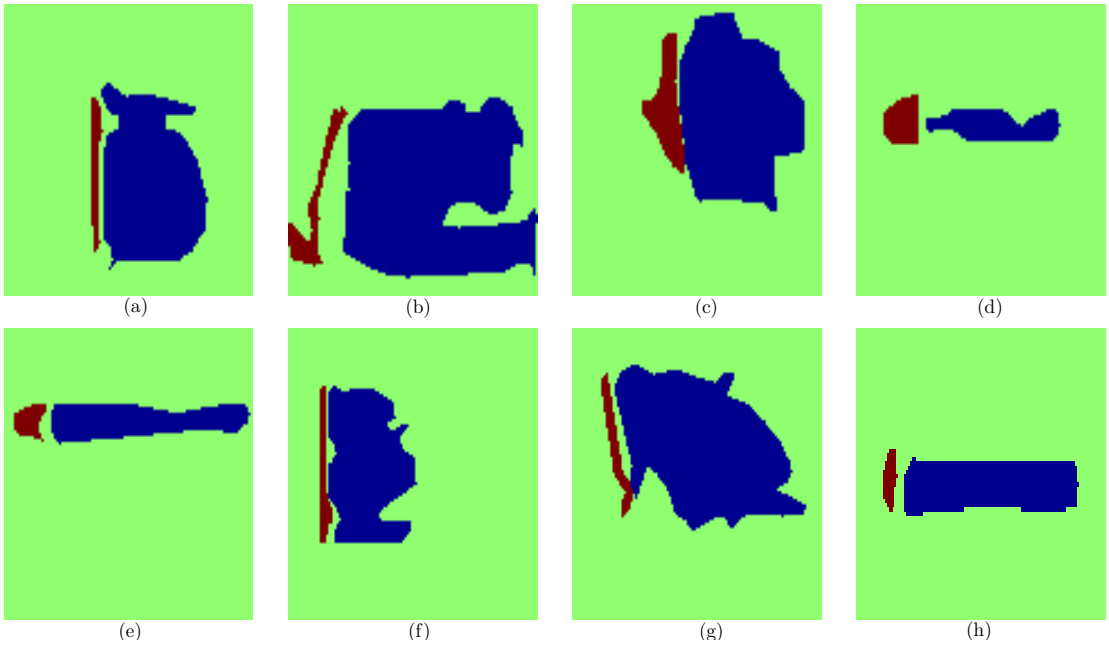


Figure 2.12. Ground truths of the images in Fig. 2.11. Subfigures (a)−(b) denote the ground truth of test image 1 to test image 8.

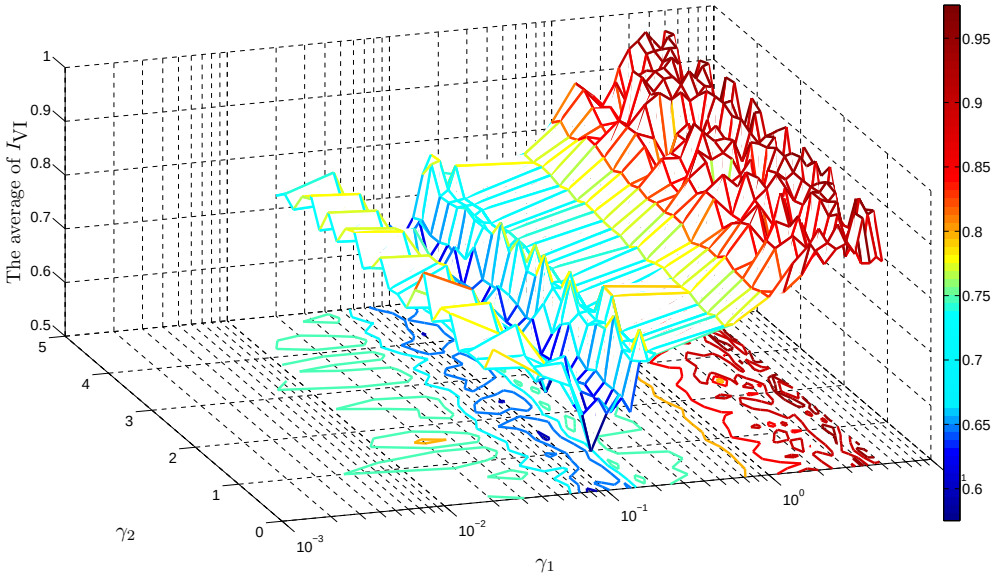


Figure 2.13. The averages of the I_{VI} over the eight test images in Fig. 2.11.

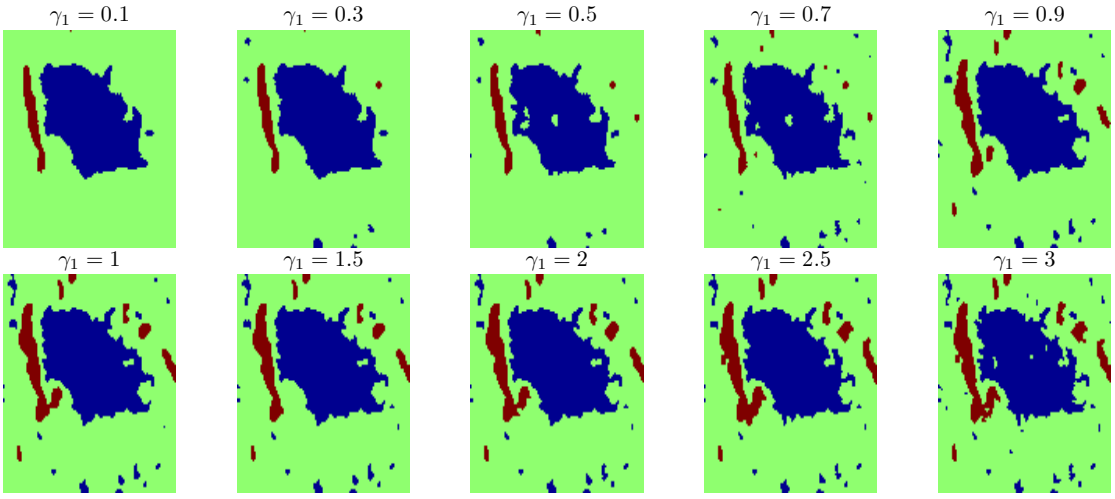


Figure 2.14. An example to illustrate the impact of γ_1 on the segmentation results. γ_2 is set to 1.

Finally, we visualize the comparison of segmentation results in Fig. 2.15. The optimal parameter setting for E-DS-M obtained in the numerical test is applied, i.e. $\gamma_1 = 0.1$ and $\gamma_2 = 1.4$. It is apparent in Fig. 2.15 that the results given by E-DS-M can provide more precise segmentation results with less mislabeled pixels than other methods.

2.5.3 Experiments on Synthetic Images

The performance of E-DS-M on SAS images with ripple-like sediments is studied in this subsection. There is a synthetic image whose dimensions are 300×300 pixels, and it contains cylinder mines. The object region and background are initially synthesized separately. According to our empirical study, the gamma distribution can be used to approximate the statistics of the pixel intensities of highlights and shadows in SAS images. The mean values and standard deviations of the gamma distributions chosen for objects are $\mu_{\text{highlight}} = 120, \sigma_{\text{highlight}} = 10$, and $\mu_{\text{shadow}} = 10, \sigma_{\text{highlight}} = 5$, respectively. The ripple sediment is simulated as given in [115]. This simulated sediment is slightly corrupted by speckle noise. Finally, we superimpose the object region and sediment as follows

$$u_{\text{syn}} = 0.8u_{\text{object}} + 0.2u_{\text{ripple}}. \tag{2.75}$$

Hence, the object regions in the resulting images are only approximately gamma distributed, since it also contains part of the sediment statistics.

The same parameter setting of γ_1 and γ_2 as in Fig. 2.15 is applied to the test on synthetic images. The results are shown in Fig. 2.16. Comparing the results of E-DS-M with those of MAP-ISO, DEM and MAP-Reed, it can be observed that E-DS-M can suppress the influence of a ripple-like sediment very well. The segmentation result is almost identical to the ground truth. Thus, it is verified that E-DS-M is also reliable when objects are lying on ripple-like sediments.

2.5.4 Computational Cost

The computational cost of the segmentation methods, i.e. MAP-ISO, MAP-Reed, DEM and E-DS-M, should also be studied. The SAS image snapshots with different sizes have been employed. Only squared snapshots of the SAS imagery are considered. The test image for the evaluation of computational cost is depicted in Fig. 2.17. Its original size is 1000×1000 pixels. We resize it into images with different side lengths. Three of them are presented as examples in Fig. 2.17. A computer equipped with an

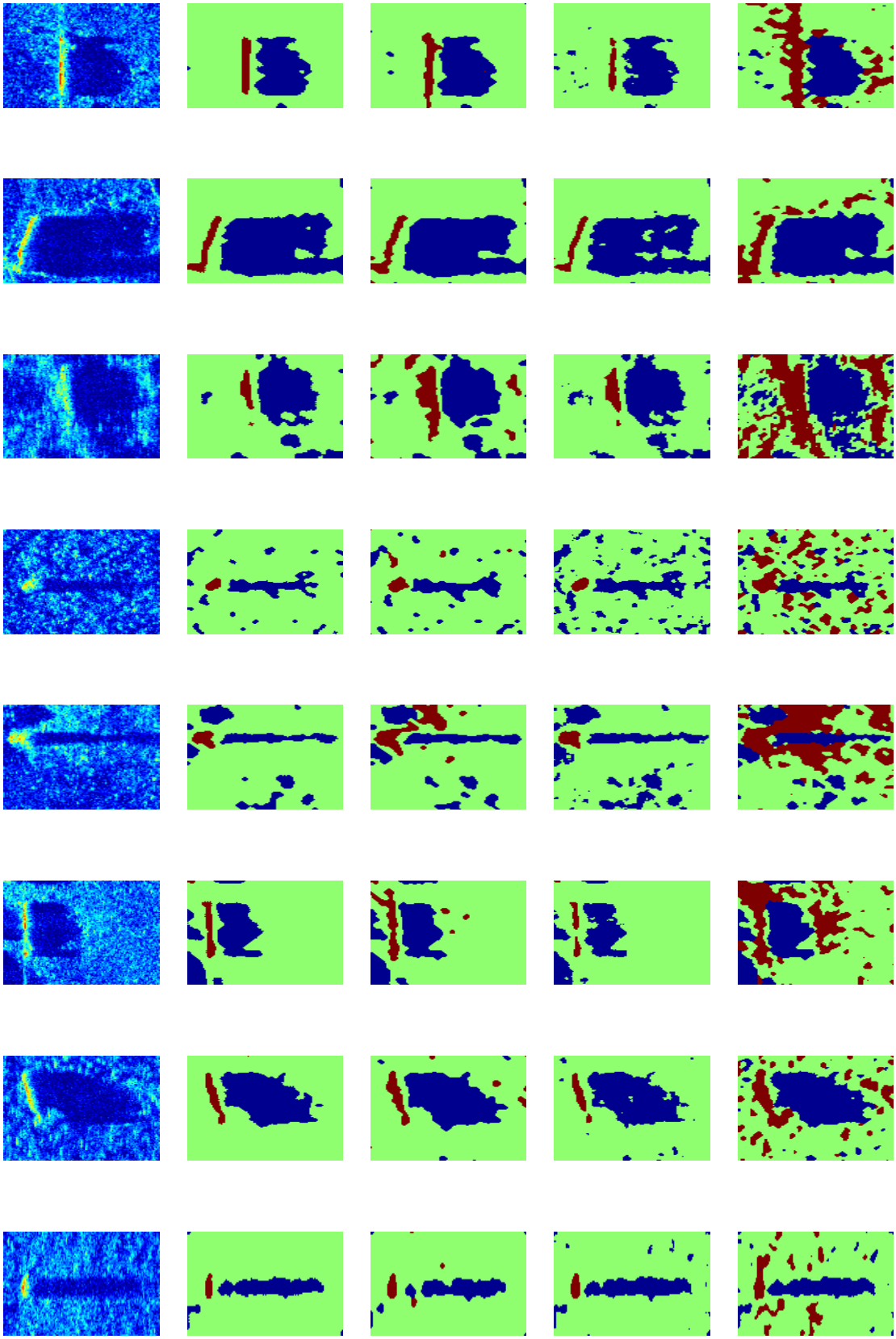


Figure 2.15. Examples of the segmentation results. Column (a) presents the sonar imagery, in column (b) up to column (e) there are segmentation results obtained by the methods E-DS-M, MAP-ISO, DEM and MAP-Reed, respectively.

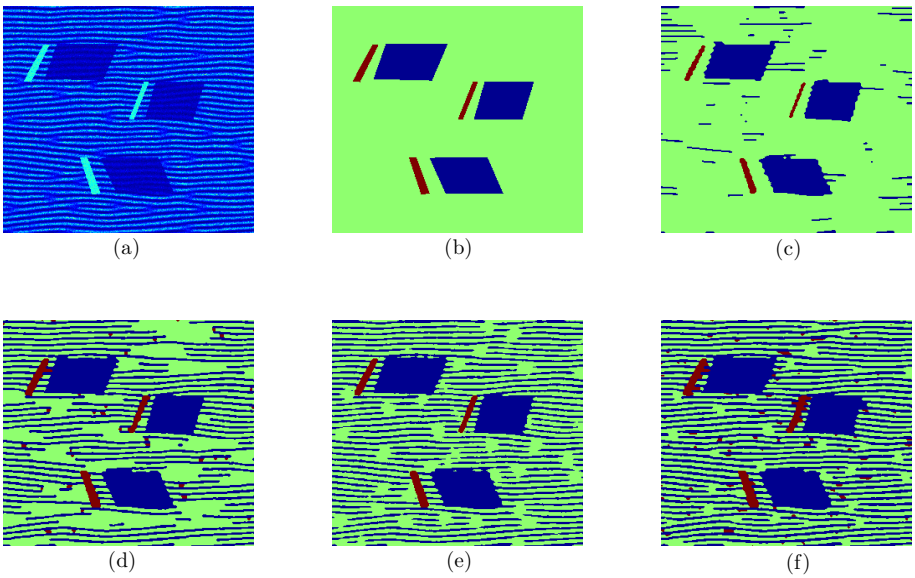


Figure 2.16. The numerical test on synthetic image: (a) synthetic image with ripple-like sediment, (b) ground truth, (c)−(d) provide the segmentation results given by E-DS-M, MAP-ISO, DEM and MAP-Reed, respectively.

Intel(R) Xeon(R) 2.93GHz processor is employed. The programs are written in Matlab. All the four methods are iterative. Hence, the computational time depends on their iteration numbers. The maximum iteration number of individual methods is set to 200. The diffusion iteration of DEM is set to 50, which is an empirical value obtained in our study so that DEM can provide a good segmentation result. Although the time required for every iteration in E-DS-M is high, it is still an efficient approach since it requires fewer iterations, i.e. usually fewer than 50 iterations. In contrast, MAP-ISO and MAP-Reed often need more than 100 iterations before the convergence is reached.

As demonstrated in Fig. 2.18, the image sizes have a great impact on the computational cost. For snapshots of smaller side lengths, the difference among methods is little. The E-DS-M sometimes could require even longer processing time than the others when the image is smaller than 240×240 pixels. This can be attributed to the fact that all the four methods require only a few iterations for images with small sizes before reaching convergence. The adoption of E-DS-M is then not very profitable. With the increasing of the image size, the advantage of choosing E-DS-M can be observed. There are several locations on the curves of MAP-Reed and E-DS-M where the computational cost is no longer increasing functions of the image size. This can be explained as follows. The time required for the neighborhood configuration histogramming in MAP-Reed and

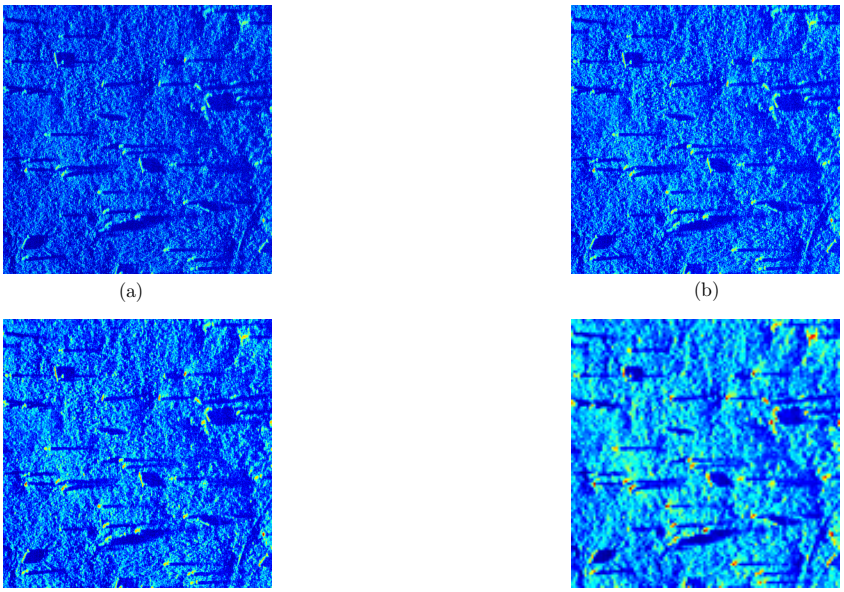


Figure 2.17. The test images used for the study of computational cost. The original image is on the top left. There are three examples of the resized images in the following. (a): side length = 1000 pixels. (b): side length = 720 pixels. (c): side length = 420 pixels. (d): side length = 120 pixels.

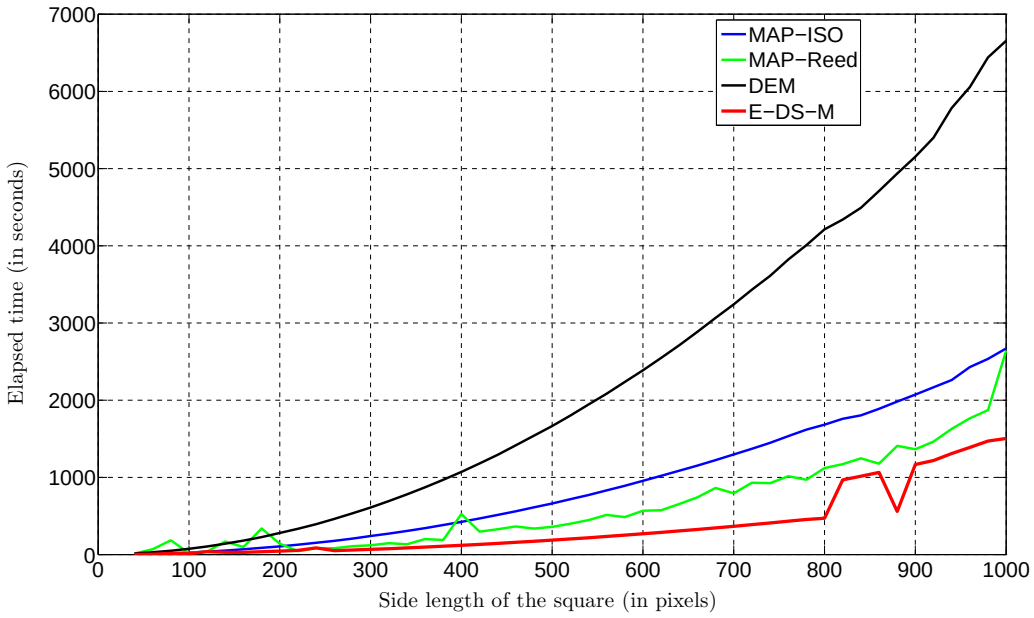


Figure 2.18. The processing time of the image snapshots with increasing size.

the combination of BBAs in E-DS-M using Equations (2.63)−(2.67) is not a strictly increasing function of the image size. One depends on how many different cases of the neighborhood configurations exist in the image, and the other is correlated to the complexity of the neighborhood configuration.

2.6 Conclusions

In this chapter, an expectation-maximization approach for image segmentation is considered. This approach is utilized to obtain the shape information of mine-like objects. The segmentation results are sent to the subsequent step of feature extraction for the extraction of geometrical features.

A generalized mixture model is employed in this expectation-maximization approach, in which the Pearson system is taken into account. Consequently, the generalized mixture model can better approximate the statistics of synthetic aperture sonar imagery than those conventional models, e.g. Gaussian mixture model. Moreover, the Dempster-Shafer theory has been incorporated to describe the correlation between neighboring pixels. A belief structure based on the pixel intensities has been proposed to quantify the dependency between pixels. We developed an iterative approach called E-DS-M for image segmentation by introducing the Dempster-Shafer clustering between each E- and M-step. The proposed approach has been applied to the synthetic aperture sonar images.

Compared with the methods in the literature, the proposed approach can considerably enhance the quality of the segmentation results. The quantitative analysis of the segmentation results shows that the E-DS-M can provide segmentation results with higher accuracy and it also demonstrates another fact that E-DS-M is only sensitive to the setting of one parameter. Therefore, it is reasonable to reduce the number of parameters involved in the Dempster-Shafer theory based clustering to one. The optimal setting for parameters is obtained in numerical tests. Besides, the study of computational cost demonstrates that E-DS-M is very efficient with the increasing of image size.

Chapter 3

Feature Extraction in Sonar Imagery

This chapter handles the feature extraction. It takes the images of the ROI and the segmentation results obtained in the step of image segmentation as its input to extract the texture features as well as the geometrical features. The extraction of features can be divided into two different phases, the *system design phase* and *object classification phase*. During the system design phase, a large number of features that are probably useful for the classification of underwater objects are extracted. Due to the curse of dimensionality (cf. Fig. 1.3), only a small part of the features are considered in the phase of object classification. The choice of relevant features, i.e. feature selection, is executed in the system design phase and its results are used to guide the feature extraction in the object classification phase so that only those relevant features will be extracted. We will introduce all the features considered in the system design phase in this chapter.

The results of MLO detection provide a database with M pieces of MLOs. Every MLO can be represented by a vector, e.g. the vector of m-th MLO is $\boldsymbol{\chi}^{(m)} = (\chi_{1,m}, ..., \chi_{n,m}, ..., \chi_{N_O,m})^T$. The element $\chi_{m,n}$ for $1 \leq n \leq N_O$ and $1 \leq m \leq M$ is the m-th realization of the random variable $\mathcal{X}_n$. The random variable $\mathcal{X}_n$ is usually referred as a feature. Let the set of all features be $\mathbf{O} = \{\mathcal{X}_1, ..., \mathcal{X}_n, ..., \mathcal{X}_{N_O}\}$, and obviously we have $N_O = |\mathbf{O}|$ features.

The features used for object classification have been intensively studied in the literature, such as geometrical features in [50] and the features dedicated to NAS imagery in [51–53]. Since the presence of the object shadow is much more reliable than that of the highlight in the imagery obtained by the NAS systems, feature extraction was mainly focused on object shadows. However, this phenomenon is less remarkable for the modern SAS systems. Moreover, the object highlights provide the direct information about the object shape. Thus, it is unreasonable to exclude them in our application.

When a feature is very classification relevant, its realizations should adopt very different values for those objects belonging to different classes. Otherwise, it is considered as insignificant. However, recent research has demonstrated that even the combination of several individually insignificant features is possibly able to create a very relevant feature set [116]. An example is shown in Fig. 3.1. There are two features $\mathcal{X}_1$ and $\mathcal{X}_2$. Individually considered, neither is able to help us to distinguish the class 1 from

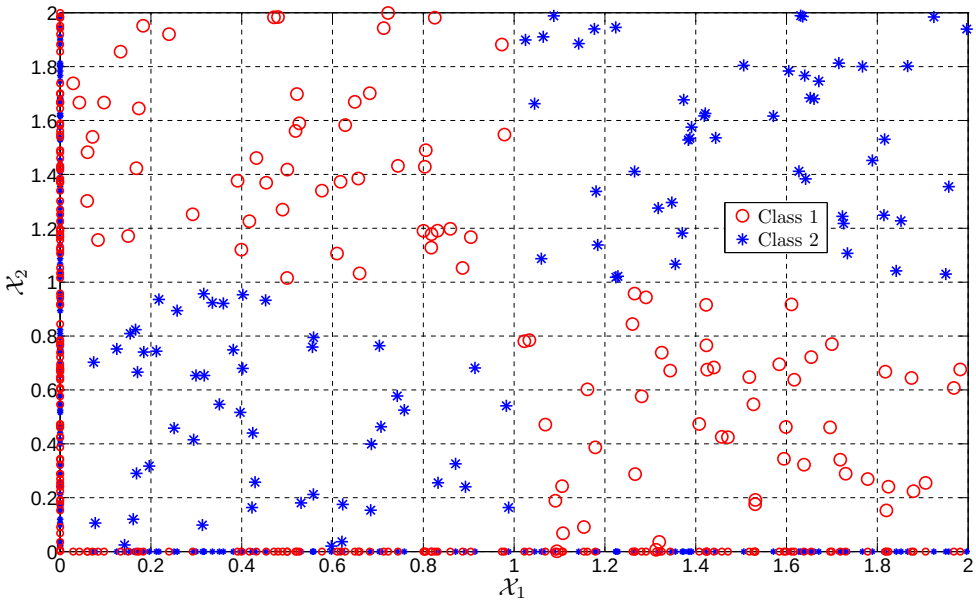


Figure 3.1. Combination of two features ($\mathcal{X}_1$ and $\mathcal{X}_2$) that are individually insignificant. Features $\mathcal{X}_1$ and $\mathcal{X}_2$ are presented along the x and y-axis, respectively. On the 2D plan constructed by these two features, the objects belonging to different classes can be easily distinguished.

the class 2, cf. the x-axis and the y-axis. There are major overlaps between objects of different classes. It is impossible to separate them into two classes with respect to either feature $\mathcal{X}_1$ or feature $\mathcal{X}_2$. However, objects belonging to different classes can be easily distinguished while jointly considering features $\mathcal{X}_1$ and $\mathcal{X}_2$. Unfortunately, the knowledge about this kind of feature combination that can dramatically improve the distinguishing ability of the features is usually unavailable *a priori*. Hence, it is practical to build a feature set with many features. In this chapter, we employ the geometrical features from [50] and invariant moments in [117] to describe the shape of the MLOs. They are applied to both the highlights and the shadows. Furthermore, we propose several novel features for our applications. The texture features in [118, 119] are included as well, since the deployment of the object on the seabed can alter the characteristics of the seabed textures.

From the segmentation results, both object regions and object contours are available. Therefore, geometrical features are divided into two subgroups: object region features and contour features. Straightforwardly, this chapter can be organized into three sections. In Secs. 3.1 and 3.2, the object region features and contour features are explained. The texture features of the ROI are listed in Sec. 3.3.

3.1 Object Region Features

The classification of underwater objects based on their geometries has been considered in the literature for a long time. Natural objects can have arbitrary shapes. Their shapes are mostly much more complex than those of man-made objects that are usually of square, circular, spherical forms and so on. Moreover, the size of man-made objects, e.g. underwater mines, lies within a certain interval. They would not be arbitrarily large or small due to the cost of production and transportation.

The length of the major and minor axes, i.e. l_{major} and l_{minor} with $l_{\text{minor}} < l_{\text{major}}$, the area of the region (A) and the extent (Extent) have been widely used as region features. The features, like l_{major}, l_{minor} and A, provide the information about the object size. The Extent of a shape is given as follows:

$$\text{Extent} = \frac{A}{A_{\text{BX}}}, \tag{3.1}$$

where A_{BX} is the area of the bounding rectangle that is the smallest rectangle enclosing the object region [120]. The Extent reaches its maximum (i.e. Extent = 1) for a rectangular object. When the object is an ideal circle, it equals to $\frac{\pi}{4}$. With the increasing of the dissimilarity to the rectangle, the Extent decreases itself. The principal axes of a given region are defined as the two line segments that cross each other orthogonally in the centroid of the region and represent the directions with zero cross-correlation [121]. The covariance matrix of a given region is given by

$$\begin{aligned} \mathbf{CM} &= \frac{1}{N_{\text{region}}} \sum_{i=1}^{N_{\text{region}}} \begin{pmatrix} x_i - x_* \\ y_i - y_* \end{pmatrix} \begin{pmatrix} x_i - x_* \\ y_i - y_* \end{pmatrix}^T \\ &= \begin{pmatrix} cm_{xx} & cm_{xy} \\ cm_{yx} & cm_{yy} \end{pmatrix}, \end{aligned} \tag{3.2}$$

where (x_*, y_*) is the centroid of the region with $x_* = \frac{1}{N_{\text{region}}} \sum_{i=1}^{N_{\text{region}}} x_i, y_* = \frac{1}{N_{\text{region}}} \sum_{i=1}^{N_{\text{region}}} y_i$ and N_{region} is the number of pixels in the object region. The lengths of the principal axes, i.e. l_{minor} and l_{major}, are equal to the two eigenvalues of the covariance matrix **CM**. So far another popular object region feature called eccentricity (Ecc) can be calculated:

$$\text{Ecc} = \frac{l_{\text{major}}}{l_{\text{minor}}}, \tag{3.3}$$

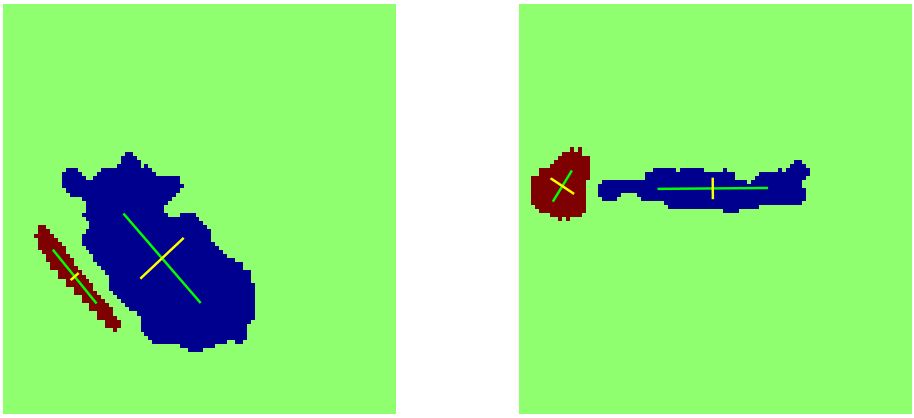
where Ecc ≥ 1. It reaches the minimum value for the shape such as square or circle and the Ecc tends to infinity as the shape approaches a straight line. Furthermore, we include the relationship between the highlight and the shadow regions as features, i.e.

area ratio (R_{area}) and axis ratio (R_{axis}). The area ratio and axis ratio are defined as

$$R_{\text{area}} = \frac{A_{\text{shad}}}{A_{\text{high}}}, \tag{3.4}$$

$$R_{\text{axis}} = \frac{l_{\text{minor,shad}}}{l_{\text{minor,high}}}, \tag{3.5}$$

where A_{shad} and A_{high} are the areas of the shadow and the highlight, respectively, and $l_{\text{minor,shad}}$ and $l_{\text{minor,high}}$ are the lengths of minor principal axes of the shadow and the highlight, respectively. In Fig. 3.2, two examples of the principal axes of object



(a) (b)

Figure 3.2. The principal axes of example objects. (a): The segmentation of a cylinder mine. (b): The segmentation of a truncated cone mine. The l_{minor} is depicted in yellow and the l_{major} is in green.

regions are presented. Along the direction of the insonifying wave, the shadows are located behind the highlights. The major axis and minor axis are depicted in green and yellow, respectively. On the left side there is a segmentation of a cylinder mine and the one of a truncated cone mine is placed on the right side. The geometry of a shadow is correlated to the geometry of its highlight, which represents the shape of the object. As shown in the figure, the width of the shadow along the direction that is orthogonal to the insonifying direction is dependent on the geometry of the object, i.e. for a cylinder mine it is correlated to the length of the cylinder and for a truncated cone mine it depends on the diameter of the truncated cone. Accordingly, the shadow of a cylinder mine is probably much greater than the one of a truncated cone mine. Consequently, the area ratio of a cylinder mine should be greater than the one of a truncated cone mine. Besides, the l_{minor} of a cylinder highlight is limited by the diameter of the cylinder mine and it is mostly much shorter than any of the principal axes of its shadow. In contrast, the l_{minor} of the truncated cone's shadow is dependent

on the diameter of the truncated cone. Although the highlight of a truncated cone is not strictly circular due to the projection, its principal axes still have similar lengths as the diameter of the truncated cone mine. Therefore, the axis ratio of a cylinder mine should be greater than the one of a truncated cone mine. The feature values of the objects in our database are illustrated in Fig. 3.3.

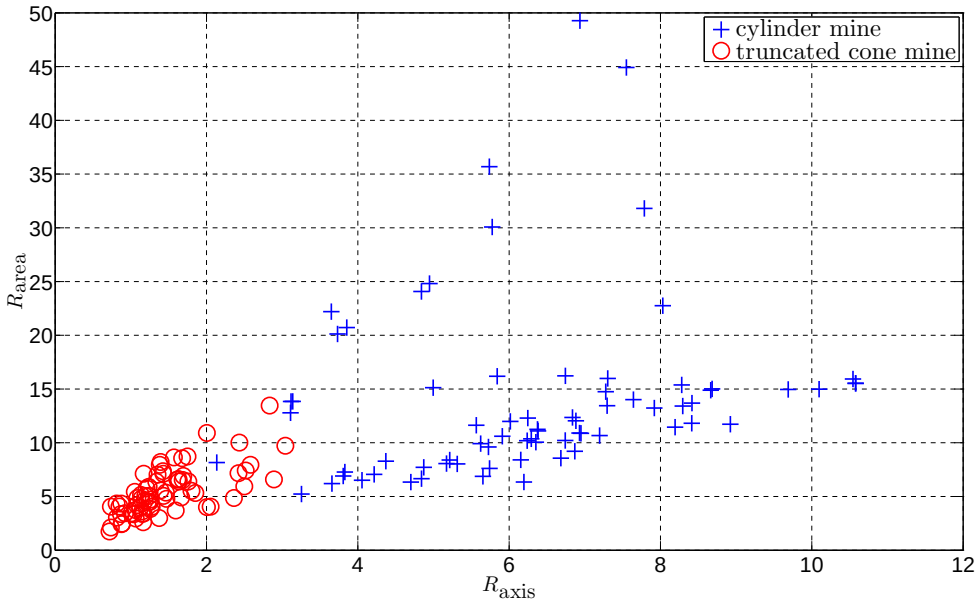


Figure 3.3. Feature values of the objects in our database, R_{area} and R_{axis}.

Recently, Tang *et al.* [122] introduced a ring projection function $f_{\text{ring}}(r)$:

$$f_{\text{ring}}(r) = \int_0^{2\pi} u_{\text{b}}(r,\theta)d\theta, \tag{3.6}$$

where $u_{\text{b}}(r,\theta)$ is a binary valued function in polar coordinates,

$$u_{\text{b}}(r,\theta) = \begin{cases} 1, & \text{if point } (r,\theta) \text{ locates in the object region} \\ 0, & \text{otherwise.} \end{cases} \tag{3.7}$$

Analogously, we propose a similar projection, the radius projection function $f_{\text{radius}}(\theta)$:

$$f_{\text{radius}}(\theta) = \int_0^{r_{\max}} u_{\text{b}}(r,\theta)dr, \tag{3.8}$$

where $r_{\max}$ is the maximum radius length in the image. In order to make the transformation scale-invariant, the normalized ring and radius projection, $\bar{f}_{\text{ring}}(r)$ and $\bar{f}_{\text{radius}}(\theta)$,

are taken into account for further computation:

$$\bar{f}_{\text{ring}}(r) = \frac{f_{\text{ring}}(r)}{\max_{r'} f_{\text{ring}}(r')}, \tag{3.9}$$

$$\bar{f}_{\text{radius}}(\theta) = \frac{f_{\text{radius}}(\theta)}{\max_{\theta'} f_{\text{radius}}(\theta')}. \tag{3.10}$$

There are examples of ring and radius projection of a strip-formed object shown in

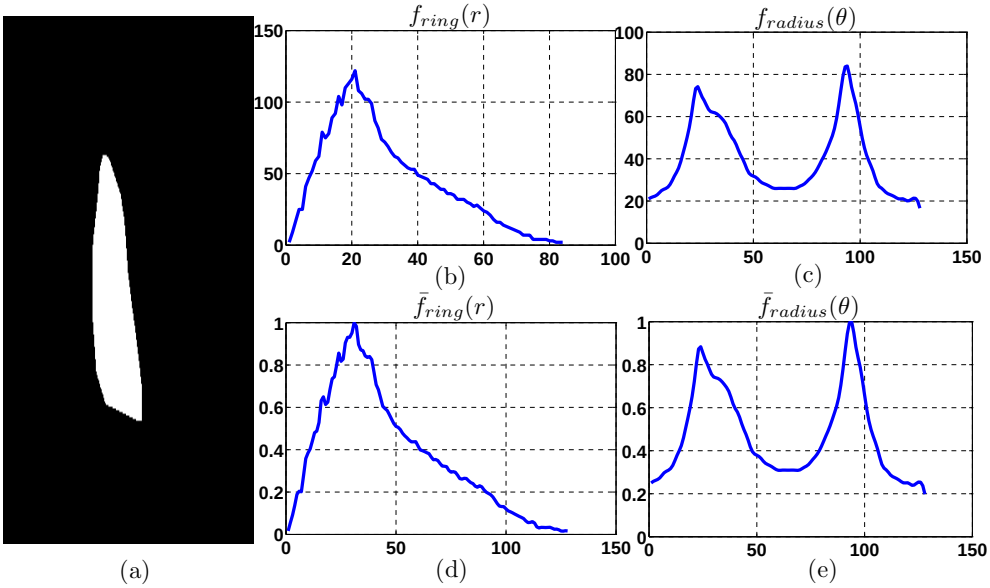


Figure 3.4. The ring and radius projections. (a): An object, (b): ring projection, (c): radius projection, (d): normalized ring projection and (e): normalized radius projection.

Fig. 3.4. Its ring projection has one peak while its radius projection has two peaks. In contrast, the ideal circular region has a linear increasing function with the slope of 2π as its ring projection and its radius projection is a constant, $\forall\theta \in [0, 2\pi]$.

In discrete case, the normalized ring and radius projection are sampled with N_{ring} and N_{radius} points, respectively. The discrete sequences of ring and radius projection are geometrical descriptors. Their dimensions are usually compressed by methods like wavelet transformation and PCA, which are out of the scope of this thesis. Thus, we extract some features based on the statistical properties of the values of $\bar{f}_{\text{ring}}(r)$ and $\bar{f}_{\text{radius}}(\theta)$: ring projection skewness ($\varepsilon_{\text{ring}}$), ring projection condensity (Den_{ring}), radius projection mean value (μ_{radius}) and radius projection skewness ($\varepsilon_{\text{radius}}$). They

are defined as follows:

$$\varepsilon_{\text{ring}} = \frac{\frac{1}{N_{\text{ring}}}\sum_{n=1}^{N_{\text{ring}}}\left(\bar{f}_{\text{ring}}(r_n)-\mu_{\text{ring}}\right)^3}{\left(\frac{1}{N_{\text{ring}}}\sum_{n=1}^{N_{\text{ring}}}\left(\bar{f}_{\text{ring}}(r_n)-\mu_{\text{ring}}\right)^2\right)^{1.5}}, \tag{3.11}$$

$$\text{Den}_{\text{ring}} = \frac{1}{N_{\text{ring}}}\sum_{n=1}^{N_{\text{ring}}}\bar{f}_{\text{ring}}(r_n)\times r_n^{0.5}, \tag{3.12}$$

$$\mu_{\text{radius}} = \frac{1}{N_{\text{radius}}}\sum_{n=1}^{N_{\text{radius}}}\bar{f}_{\text{radius}}(\theta_n), \tag{3.13}$$

$$\varepsilon_{\text{radius}} = \frac{\frac{1}{N_{\text{radius}}}\sum_{n=1}^{N_{\text{radius}}}\left(\bar{f}_{\text{radius}}(\theta_n)-\mu_{\text{radius}}\right)^3}{\left(\frac{1}{N_{\text{radius}}}\sum_{n=1}^{N_{\text{radius}}}\left(\bar{f}_{\text{radius}}(\theta_n)-\mu_{\text{radius}}\right)^2\right)^{1.5}}, \tag{3.14}$$

where $\mu_{\text{ring}} = \frac{1}{N_{\text{ring}}}\sum_{n=1}^{N_{\text{ring}}}\bar{f}_{\text{ring}}(r_n)$ is the mean value of the normalized ring projection. As discussed above, the difference in geometries is conveyed to the projection functions, and accordingly the statistical properties such as skewness and mean value are distinct. These distinctions can be clearly observed in Fig. 3.5. Due to the difficulty of displaying 4D space, we combine three out of the four features to create 3D feature spaces. It is obvious that the cylinder can be easily differentiated from the truncated cone with the help of ring and radius projection features.

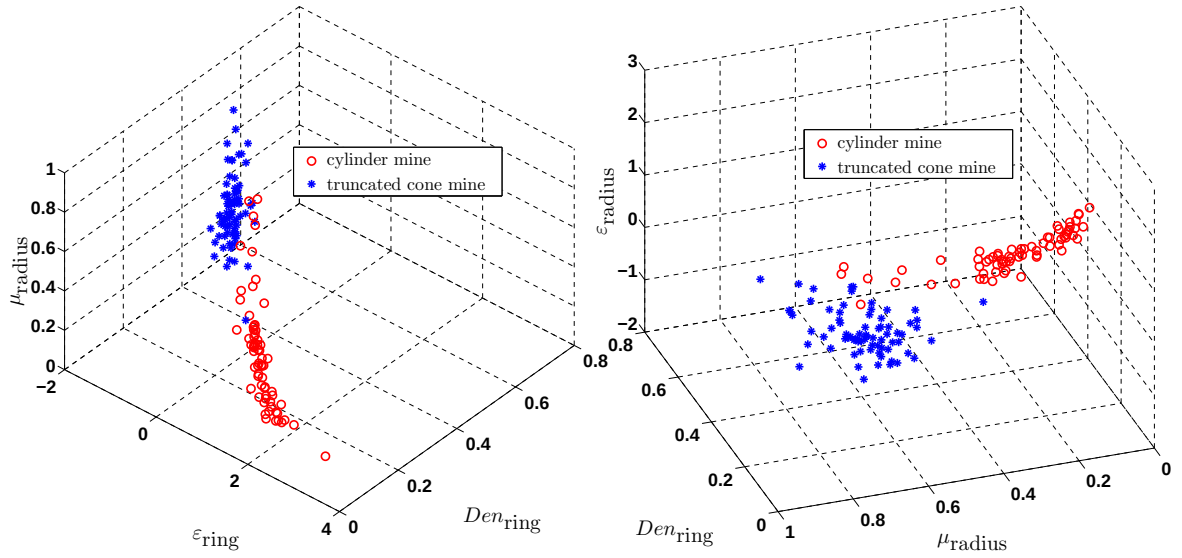


Figure 3.5. Feature values of the object highlights in our database. (a): The combination of $\varepsilon_{\text{ring}}$, Den_{ring} and μ_{radius}. (b): The combination of Den_{ring}, μ_{radius} and $\varepsilon_{\text{radius}}$.

In addition, the well-known rotation invariant moments given by Hu [117] are considered. The image moments are invariant under translation, changes in scale, and also

rotation. They consist of six absolute orthogonal invariants:

$$\mathcal{G}_1 = \tilde{\zeta}_{2,0} + \tilde{\zeta}_{0,2} \tag{3.15}$$

$$\mathcal{G}_2 = (\tilde{\zeta}_{2,0} - \tilde{\zeta}_{0,2})^2 + 4\tilde{\zeta}_{1,1}^2 \tag{3.16}$$

$$\mathcal{G}_3 = (\tilde{\zeta}_{3,0} - 3\tilde{\zeta}_{1,2})^2 + (3\tilde{\zeta}_{2,1} - \tilde{\zeta}_{0,3})^2 \tag{3.17}$$

$$\mathcal{G}_4 = (\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2})^2 + (\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3})^2 \tag{3.18}$$

$$\begin{aligned}\mathcal{G}_5 = {} & (\tilde{\zeta}_{3,0} - 3\tilde{\zeta}_{1,2})(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2}) \left[(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2})^2 - 3(\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3})^2\right] + \\ & (3\tilde{\zeta}_{2,1} - \tilde{\zeta}_{0,3})(\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3}) \left[3(\tilde{\zeta}_{2,1} + \tilde{\zeta}_{1,2})^2 - (\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3})^2\right]\end{aligned} \tag{3.19}$$

$$\begin{aligned}\mathcal{G}_6 = {} & (\tilde{\zeta}_{2,0} - \tilde{\zeta}_{0,2}) \left[(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2})^2 - (\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3})^2\right] + \\ & 4\tilde{\zeta}_{1,1}(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2})(\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3}),\end{aligned} \tag{3.20}$$

and one skew orthogonal invariant,

$$\begin{aligned}\mathcal{G}_7 = {} & (3\tilde{\zeta}_{2,1} - \tilde{\zeta}_{0,3})(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2}) \left[(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2})^2 - 3(\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3})^2\right] - \\ & (\tilde{\zeta}_{3,0} - 3\tilde{\zeta}_{1,2})(\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3}) \left[3(\tilde{\zeta}_{3,0} + \tilde{\zeta}_{1,2})^2 - (\tilde{\zeta}_{2,1} + \tilde{\zeta}_{0,3})^2\right],\end{aligned} \tag{3.21}$$

where $\tilde{\zeta}_{i,j}$ is the $(i+j)$-th order central moments of a given region [120].

3.2 Contour Features

The object contour refers to a closed curve denoting the boundary between object region and background region. There are two examples depicted in Fig. 3.6. In discrete case the contour is approximated by N_{contour} line segments, and there are N_{contour} points/vertices on the contour. Let $d_{\text{cen}}(n)$ be the centroid distance,

$$d_{\text{cen}}(n) = \sqrt{(x_n^{\mathbb{L}} - x_*^{\mathbb{L}})^2 + (y_n^{\mathbb{L}} - x_*^{\mathbb{L}})^2}, \tag{3.22}$$

where the centroid $(x_*^{\mathbb{L}}, y_*^{\mathbb{L}})$ of the contour is defined as [50]

$$x_*^{\mathbb{L}} = \frac{1}{6\mathcal{A}} \sum_{n=0}^{N_{\text{contour}}-1} \left(x_n^{\mathbb{L}} + x_{n+1}^{\mathbb{L}}\right)\left(x_n^{\mathbb{L}} y_{n+1}^{\mathbb{L}} - x_{n+1}^{\mathbb{L}} y_n^{\mathbb{L}}\right), \tag{3.23}$$

$$y_*^{\mathbb{L}} = \frac{1}{6\mathcal{A}} \sum_{n=0}^{N_{\text{contour}}-1} \left(y_n^{\mathbb{L}} + y_{n+1}^{\mathbb{L}}\right)\left(x_n^{\mathbb{L}} y_{n+1}^{\mathbb{L}} - x_{n+1}^{\mathbb{L}} y_n^{\mathbb{L}}\right), \tag{3.24}$$

where $\mathcal{A} = 0.5 \left|\sum_{n=0}^{N_{\text{contour}}-1} (x_n^{\mathbb{L}} y_{n+1}^{\mathbb{L}} - x_{n+1}^{\mathbb{L}} y_n^{\mathbb{L}})\right|$ and $(x_n^{\mathbb{L}}, y_n^{\mathbb{L}})$ is the n-th vertex on the contour $\mathbb{L}$.

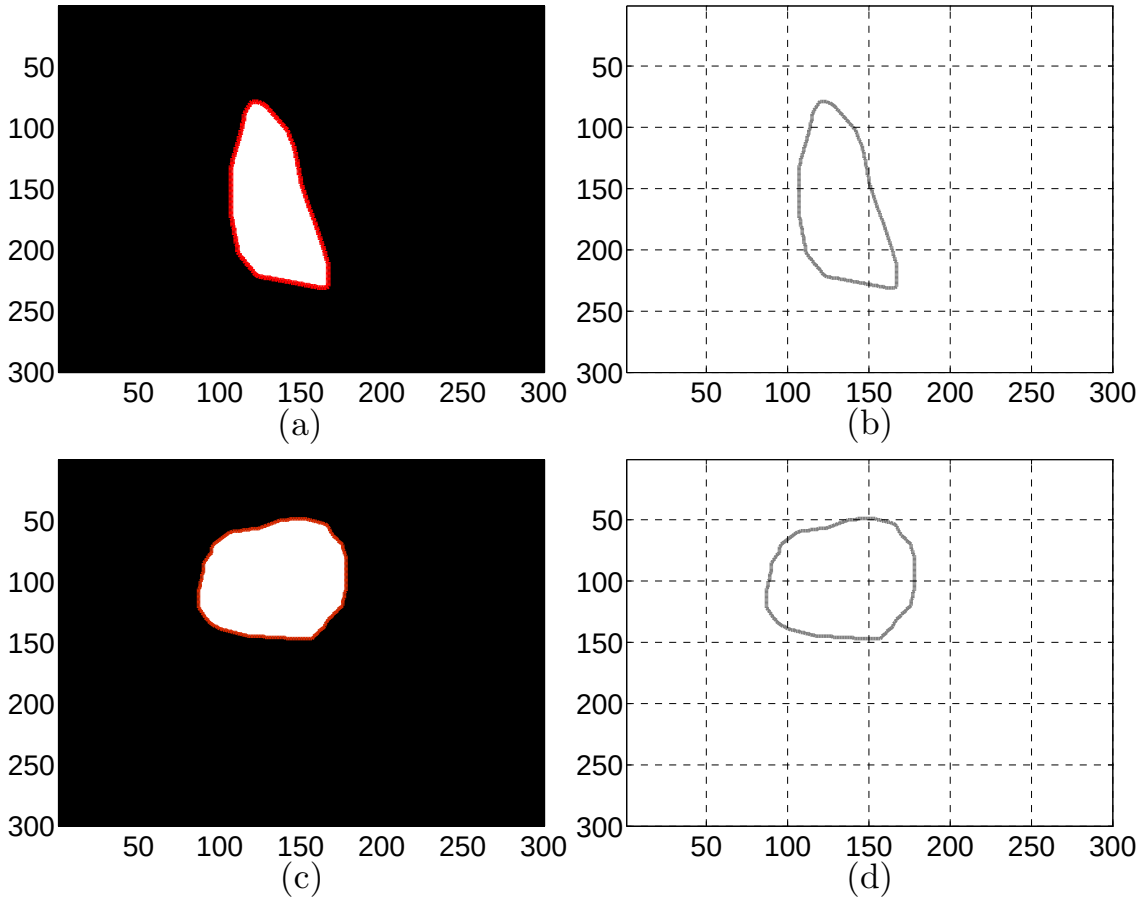


Figure 3.6. Two examples of object contours. (a) and (c): Object segmentations. The pixels inside the object region are depicted in white. The red curves are object contours. (b) and (d): The object contours in left figures are exclusively depicted in gray.

Evidently, objects with larger sizes are inclined to have longer contours. The perimeter of the contour (P_{con}) is taken into consideration as a contour feature. Another feature called compactness (Comp) is given as

$$\text{Comp} = \frac{P_{\text{con}}}{A}, \tag{3.25}$$

where A is the area within the contour. The Comp achieves its minimum for a circle and approaches infinity as the region tends to a straight line. Furthermore, features such as circularity ratio ($R_{c,1}$ and $R_{c,2}$),

$$R_{c,1} = \frac{A}{A_c}, \tag{3.26}$$

$$R_{c,2} = \frac{A}{P_{\text{con}}^2}, \tag{3.27}$$

where A_c is the area of the circle having the same length as the perimeter of the contour, circle variance (R_{va}),

$$R_{\text{va}} = \frac{\sigma_d}{\mu_d} \tag{3.28}$$

where σ_d and μ_d are the mean and standard deviation of the centroid distance d_{cen}, and solidity (Sol) of the contour in [50]

$$\text{Sol} = \frac{A}{A_{\text{convex hull}}}, \tag{3.29}$$

where $A_{\text{convex hull}}$ is the area of the convex hull [121] (cf. Fig. 3.7), are also included in our feature set $\mathbf{O}$.

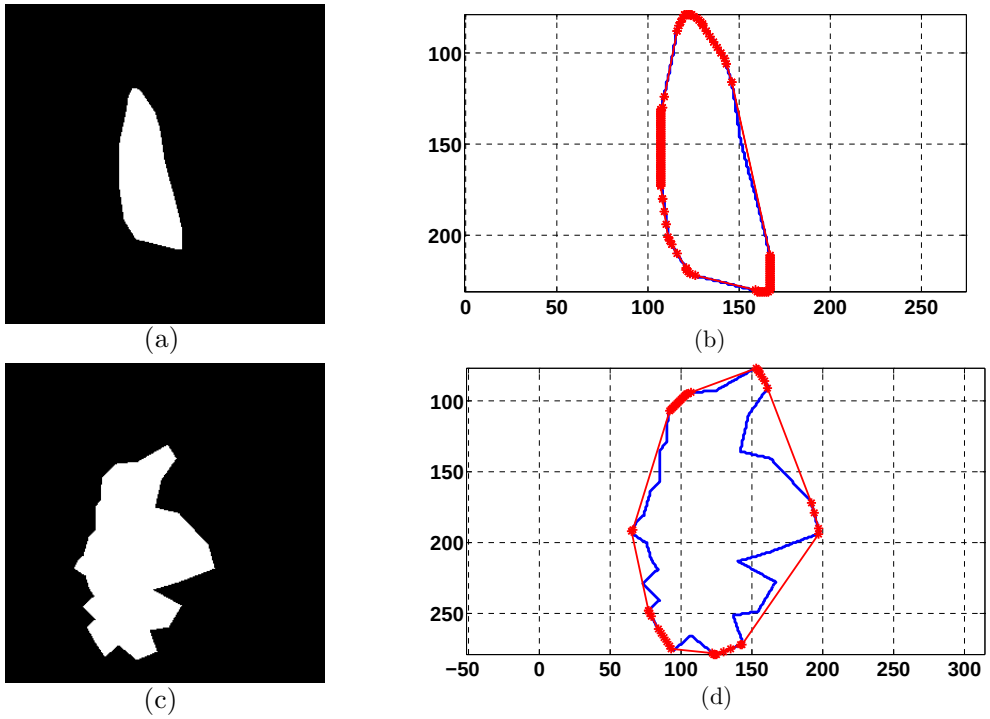


Figure 3.7. Convex hulls of objects. (a) and (c): Object segmentations. (b) and (c): Object contours (in blue) and their convex hulls (in red). (b) denotes the contour of object 1, and (d) is the contour of object 2.

The measures characterizing the smoothness of object contours can be used as features to describe objects such as the case depicted in Fig. 3.7. Due to production cost, a man-made object is most likely to have common shapes, e.g. circles and squares. The case of natural objects is probably much more complicated. Their shapes are expected to be arbitrary. As in the case of object 2 in Fig. 3.7, because of the frequent transition between concave and convex shapes, the convex hull can hardly approximate the shape of this contour. We propose the roughness ($\mathcal{V}$) of the contour, degree of curving (DoC) and absolute curvature mean value (κ_{mean}) as contour features.

The roughness ($\mathcal{V}$) is defined as

$$\mathcal{V} = \frac{P_{\text{con}}}{P_{\text{convex hull}}}, \tag{3.30}$$

where P_{con} and $P_{\text{convex hull}}$ are the perimeters of object contour and convex hull, respectively. The straight line is the shortest way between two points. Therefore, the $P_{\text{convex hull}}$ of object 2 is much shorter than its P_{con}. The roughness approaches infinity when the object contour becomes unlimited rough. This $\mathcal{V}$ can also achieve a large value when the shape is smooth but concave, e.g. a crescent. However, there is a limitation for the feature value in this case. It cannot be arbitrarily large. Taking the case in Fig. 3.7 as an example, the $\mathcal{V}$ of object 1 equals to 1.21 and the one of object 2 is 1.42. The κ_{mean} is defined as

$$\kappa_{\text{mean}} = \frac{1}{N_{\text{contour}}} \sum_{n=1}^{N_{\text{contour}}} |\kappa_n|, \tag{3.31}$$

where κ_n is the contour curvature at n-th vertex,

$$\kappa_n = \frac{\dot{x_n}^{\mathbb{L}}\ddot{y_n}^{\mathbb{L}} - \dot{y_n}^{\mathbb{L}}\ddot{x_n}^{\mathbb{L}}}{\left((\dot{x_n}^{\mathbb{L}})^2 + (\dot{y_n}^{\mathbb{L}})^2\right)^{1.5}}, \tag{3.32}$$

where $\dot{x_n}^{\mathbb{L}}$, $\dot{y_n}^{\mathbb{L}}$ are the first order derivatives, and $\ddot{x_n}^{\mathbb{L}}$, $\ddot{y_n}^{\mathbb{L}}$ are the second order derivatives. If an object contour is smooth, there could only be a few points with large value of $|\kappa_n|$ on it. The value of κ_{mean} will be small. The DoC quantifying the curving of a contour is a weighted average of the absolute curvature values,

$$\text{DoC} = \frac{\sum_{n=1}^{N_{\text{contour}}} d_n |\kappa_n|}{P_{\text{con}}}, \tag{3.33}$$

where d_n is the length of n-th line segment on the contour. The DoC describes the

	$\mathcal{V}$	κ_{mean}	DoC
Object 1	1.21	0.0099	0.5075
Object 2	1.42	0.0197	0.6140

Table 3.1. Feature values of the contours depicted in Fig. 3.7.

curving of the complete contour. The curvature values are weighted by the curve lengths so that only those curves that are mostly highly curved can possess a great value of DoC, e.g. the DoC of object 2 is greater than the one of object 1. The feature values of the samples in Fig. 3.7 are summarized in Table 3.1. Furthermore, the feature values extracted from the objects in our database are presented in Fig. 3.8. The difference between cylinder mines, truncated cone mines and rocks can be clearly characterized by the combination of these three features.

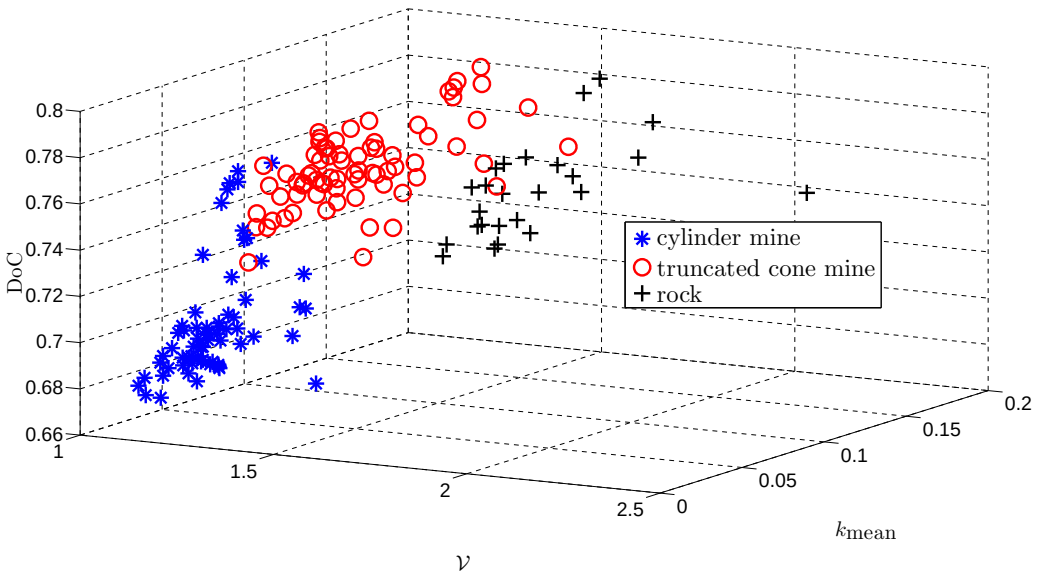


Figure 3.8. Feature values of the object highlights in our database, $\mathcal{V}$, κ_{mean} and DoC.

The Fourier descriptor is also widely adopted to specify the object's geometry. Let

$$D_{\text{cen}}(n_{\text{DFT}}) = \left| \sum_{k=0}^{N_{\text{DFT}}-1} \tilde{d}_{\text{cen}}(k) e^{-j\frac{2\pi k}{N_{\text{DFT}}} n_{\text{DFT}}} \right|, \tag{3.34}$$

$$\text{with } \tilde{d}_{\text{cen}}(k) = d_{\text{cen}}(k) - \frac{1}{N_{\text{DFT}}} \sum_{k'=0}^{N_{\text{DFT}}-1} d_{\text{cen}}(k')$$

be the magnitude of the Fourier coefficients of the centroid distance function. We implement an N_{DFT}-point discrete Fourier transform (DFT). There are examples of the Fourier descriptor depicted in Fig. 3.9. On the left there is a strip formed region, and the right one is approximately circular. For clarity, the direct current (DC) component is removed. Both of them condense their energy in the low frequency region. Since the circular form in the top right of Fig. 3.9 loses most of its energy while removing the DC component, its D_{cen} is less significant than that of the strip form. Similar as the case discussed with ring and radius projection, the sequence of D_{cen} will not be used as a shape descriptor. We propose instead two features characterizing the difference in the statistical properties of D_{cen}, i.e. low frequency density (ϱ_{LF}) and Fourier coefficient skewness (ε_{DFT}).

$$\varrho_{\text{LF}} = \frac{1}{N_{\text{LF}}} \sum_{n_{\text{DFT}}=1}^{N_{\text{LF}}} D_{\text{cen}}(n_{\text{DFT}}), \tag{3.35}$$

$$\varepsilon_{\text{DFT}} = \frac{\frac{1}{N_{\text{DFT}}} \sum_{n_{\text{DFT}}=1}^{N_{\text{DFT}}} \left(D_{\text{cen}}(n_{\text{DFT}}) - \mu_{\text{cen}}\right)^3}{\left(\frac{1}{N_{\text{DFT}}} \sum_{n_{\text{DFT}}=1}^{N_{\text{DFT}}} \left(D_{\text{cen}}(n_{\text{DFT}}) - \mu_{\text{cen}}\right)^2\right)^{1.5}}, \tag{3.36}$$

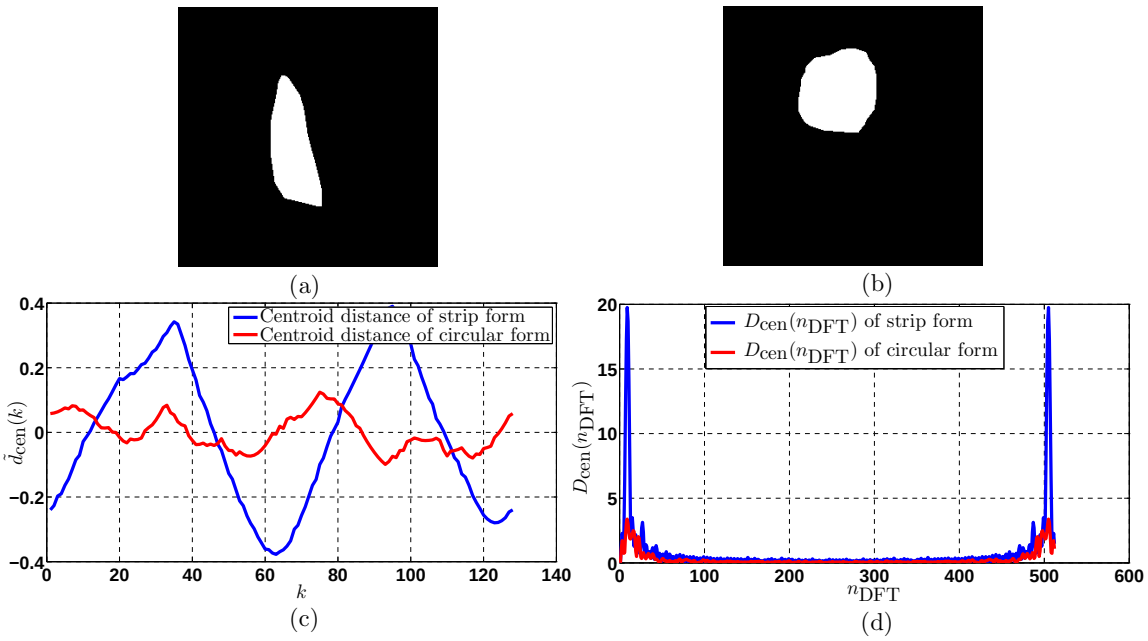


Figure 3.9. (a) and (b): Object segmentations, (c): centroid distances and (d): the magnitude of their Fourier coefficients.

where $\mu_{\text{cen}} = \frac{1}{N_{\text{DFT}}} \sum_{n_{\text{DFT}}=1}^{N_{\text{DFT}}} D_{\text{cen}}(n_{\text{DFT}})$, and $N_{\text{LF}} < N_{\text{DFT}}$ denotes the low frequency boundary index. As already discussed, the ϱ_{LF} of the strip is greater than that of a circle. Hence, it is a proper feature to distinguish a cylinder from a truncated cone. Moreover, the histogram of D_{cen} of a strip is inclined to have a longer tail due to the significant components in the low frequency band as shown in the bottom right of Fig. 3.9. This difference can be captured by ε_{DFT}. The feature values extracted from the cylinder mines and truncated cone mines are depicted in Fig. 3.10.

All of the above-mentioned geometrical features are summarized in Table 3.2. Except the R_{area} and R_{axis}, the geometrical features are applied to both the highlights and the shadows. Therefore, we have a total of 56 geometrical features in the feature set $\mathbf{O}$.

3.3 Texture Features

The texture refers to the repeating patterns of the local variation of pixel intensities. It has been applied to the problems of remote sensing to classify radar imagery into different regions, e.g. forests, lakes and residential districts. In underwater acoustics, there are different types of seabed, e.g. the flat bottom and ripple-like bottom as shown in Fig. 3.11. They are able to be distinguished by texture features.

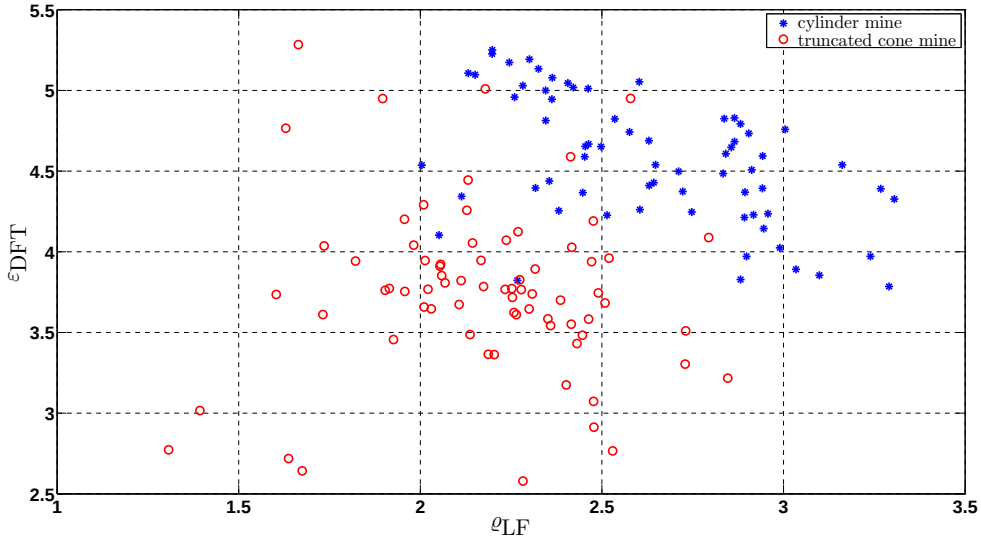
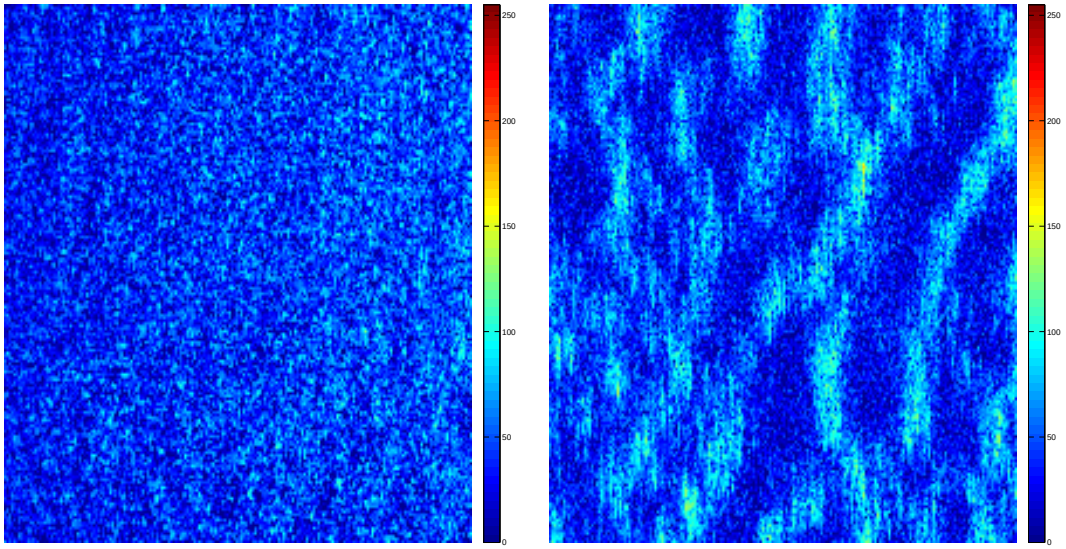


Figure 3.10. Feature values of the object highlights in our database, ϱ_{LF} and $\varepsilon_{\mathrm{DFT}}$.

Feature	**Description**
l_{major}	length of the major axis of a given region
l_{minor}	length of the minor axis of a given region
Extent	extent of a given region
A	area of a given region
$\mathcal{G}_i$	the seven Hu's invariant moments for $i = 1, ..., 7$
$\varepsilon_{\mathrm{ring}}$	ring projection skewness
$\mathrm{Den}_{\mathrm{ring}}$	ring projection condensity
μ_{ring}	radius projection mean value
$\varepsilon_{\mathrm{radius}}$	radius projection skewness
R_{area}	area ratio
R_{axis}	axis ratio
P_{con}	perimeter of a given contour
Comp	compactness of a given contour
Ecc	eccentricity of a given region
$R_{c,1}$, $R_{c,2}$	circularity ratios of a given contour
R_{va}	circle variance
Sol	solidity of a given contour
DoC	degree of curving
κ_{mean}	absolute curvature mean value of a given contour
$\mathcal{V}$	roughness of a given contour
ϱ_{LF}	low frequency density
$\varepsilon_{\mathrm{DFT}}$	Fourier coefficient skewness

Table 3.2. Summary of the geometrical features.



(a) (b)

Figure 3.11. Two types of seabed in the sonar imagery, (a): flat bottom and (b): ripple-like bottom.

Furthermore, our study finds that the presence of MLOs can change the texture characteristics of the seabed. This change is dependent on the type of MLO, for instance a cylinder has a heavier impact than a truncated cone because its shadow covers a larger area. Hence, the texture is applicable to the MLO classification. In this thesis, the co-occurrence matrix (COOC) and gray level run length matrix (GLRL), which have recently been recognized as standard features for texture classification, are employed by us.

The COOC matrix is defined over an image to be the distribution of co-occurring values at a given offset. In this chapter, let matrix $\tilde{\mathbf{u}}$ denote the 2D image with the dimension of $N_x \times N_y$, the pixel intensities of $\tilde{\mathbf{u}}$ are integers, and let $\mathbb{U}$ be the set of all possible states of pixel intensities in $\tilde{\mathbf{u}}$ and $N_g = |\mathbb{U}|$. Mathematically, a co-occurrence matrix $\boldsymbol{\mathcal{B}}$ is defined over $\tilde{\mathbf{u}}$, parametrized by an offset (d_x, d_y),

$$\begin{aligned} \boldsymbol{\mathcal{B}}(i,j|d_x,d_y) &= \#\{((n_x,n_y),(n_{x'},n_{y'})) \in (L_x \times L_y) \times (L_x \times L_y)| \\ &\quad n_x - n_{x'} = d_x, n_y - n_{y'} = d_y, \tilde{\mathbf{u}}(n_x,n_y) = i, \tilde{\mathbf{u}}(n_{x'},n_{y'}) = j\}, \end{aligned} \tag{3.37}$$

where $L_x = \{1,2,...,N_x\}, L_y = \{1,2,...,N_y\}$, and $i,j \in \mathbb{U}$, # denotes the number of elements in the set. Its dimension is of $N_g \times N_g$. The offset (d_x, d_y) controls pixel pairs in four spatial configurations: $0°(d_y = 0, d_x \neq 0)$, $45°(d_y = -d_x)$, $90°(d_x = 0, d_y \neq 0)$ and $135°(d_x = d_y)$, which are also illustrated in Fig. 3.12. Hence, the $\boldsymbol{\mathcal{B}}$ does not only depend on the distance between pairs of pixels but also their relative spatial positions.

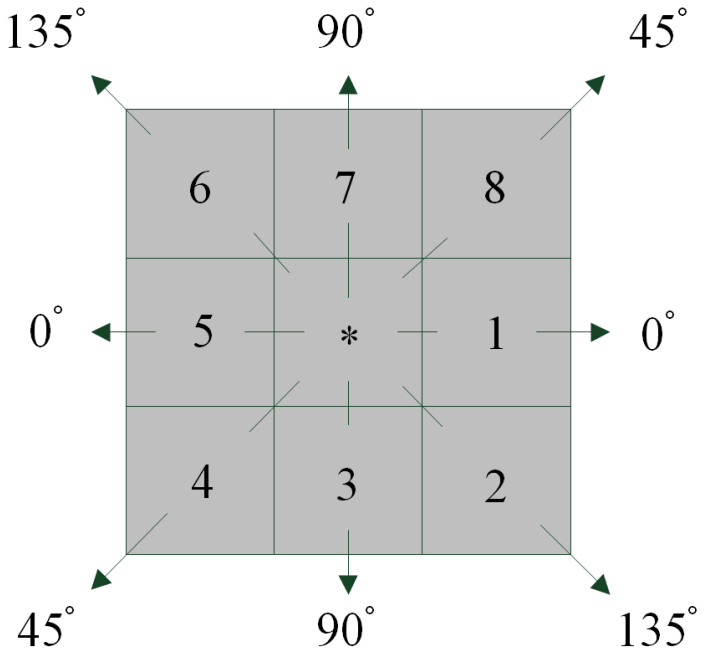


Figure 3.12. Spatial configurations of the pixel pairs: Pixels 1 and 5 construct 0° pairs with pixel $*$; pixels 2 and 6 construct 135° pairs with pixel $*$; pixels 3 and 7 construct 90° pairs with pixel $*$; and pixels 4 and 8 construct 45° pairs with pixel $*$.

There are many features defined over COOC in the literature [118, 123]. According to our studies in the sonar imagery, 12 features out of them are chosen. We define

$$\bar{\boldsymbol{\mathcal{B}}}(i,j) = \frac{\boldsymbol{\mathcal{B}}(i,j)}{\sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}\boldsymbol{\mathcal{B}}(i,j)}, \tag{3.38}$$

$$\bar{\boldsymbol{\mathcal{B}}}_i(i) = \sum_{j\in\mathbb{U}}\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.39}$$

$$\bar{\boldsymbol{\mathcal{B}}}_j(j) = \sum_{i\in\mathbb{U}}\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.40}$$

$$\tag{3.41}$$

and the features are given as follows:

- Angular second moment (ASD)

$$\text{ASD} = \sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}\bar{\boldsymbol{\mathcal{B}}}(i,j)^2, \tag{3.42}$$

- Inertia

$$\text{Inertia} = \sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}(i-j)^2\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.43}$$

- Correlation

$$\text{Correlation} = \frac{\sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}(ij)\bar{\boldsymbol{\mathcal{B}}}(i,j) - \mu_{\bar{\boldsymbol{\mathcal{B}}}_i}\mu_{\bar{\boldsymbol{\mathcal{B}}}_j}}{\sigma_{\bar{\boldsymbol{\mathcal{B}}}_i}\sigma_{\bar{\boldsymbol{\mathcal{B}}}_i}}, \tag{3.44}$$

  where $\mu_{\bar{\boldsymbol{\mathcal{B}}}_i}, \mu_{\bar{\boldsymbol{\mathcal{B}}}_j}, \sigma_{\bar{\boldsymbol{\mathcal{B}}}_i}$ and $\sigma_{\bar{\boldsymbol{\mathcal{B}}}_j}$ are the means and standard deviations of $\bar{\boldsymbol{\mathcal{B}}}_i$ and $\bar{\boldsymbol{\mathcal{B}}}_j$, respectively,

- Entropy

$$\text{Entropy} = -\sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}\bar{\boldsymbol{\mathcal{B}}}(i,j)\log_2\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.45}$$

- Shade

$$\text{Shade} = \sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}(i+j-\mu_{\bar{\boldsymbol{\mathcal{B}}}_i}-\mu_{\bar{\boldsymbol{\mathcal{B}}}_j})^3\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.46}$$

- Inverse Difference Moment (IDM)

$$\text{IDM} = \sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}\frac{1}{1+(i-j)^2}\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.47}$$

- Promenance

$$\text{Promenance} = \sum_{i\in\mathbb{U}}\sum_{j\in\mathbb{U}}(i+j-\mu_{\bar{\boldsymbol{\mathcal{B}}}_i}-\mu_{\bar{\boldsymbol{\mathcal{B}}}_j})^4\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.48}$$

- Sum Average (SA)

$$\text{SA} = \sum_{\mathfrak{i}\in\breve{\mathbb{U}}}\mathfrak{i}\bar{\boldsymbol{\mathcal{B}}}_{i+j}(\mathfrak{i}), \tag{3.49}$$

  where

$$\bar{\boldsymbol{\mathcal{B}}}_{i+j}(\mathfrak{i}) = \sum_{i\in\mathbb{U}}\sum_{\substack{j\in\mathbb{U}\text{ and}\\ i+j=\mathfrak{i}}}\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.50}$$

  where $\mathfrak{i}\in\breve{\mathbb{U}} = \{i+j | i,j\in\mathbb{U}\}$,

- Sum Entropy (SE)

$$\text{SE} = -\sum_{\mathfrak{i}\in\breve{\mathbb{U}}}\bar{\boldsymbol{\mathcal{B}}}_{i+j}(\mathfrak{i})\log_2\bar{\boldsymbol{\mathcal{B}}}_{i+j}(\mathfrak{i}) \tag{3.51}$$

- Sum Variance (SV)

$$\text{SV} = \sum_{\mathfrak{i}\in\breve{\mathbb{U}}}(\mathfrak{i}-SE)^2\bar{\boldsymbol{\mathcal{B}}}_{i+j}(\mathfrak{i}) \tag{3.52}$$

- Difference Variance (DV)

$$\bar{\boldsymbol{\mathcal{B}}}_{i-j}(\mathfrak{j}) = \sum_{i\in\mathbb{U}}\sum_{\substack{j\in\mathbb{U}\text{ and}\\ |i-j|=\mathfrak{j}}}\bar{\boldsymbol{\mathcal{B}}}(i,j), \tag{3.53}$$

$$\text{DV} = \text{variance of } \bar{\boldsymbol{\mathcal{B}}}_{i-j}(\mathfrak{j}), \tag{3.54}$$

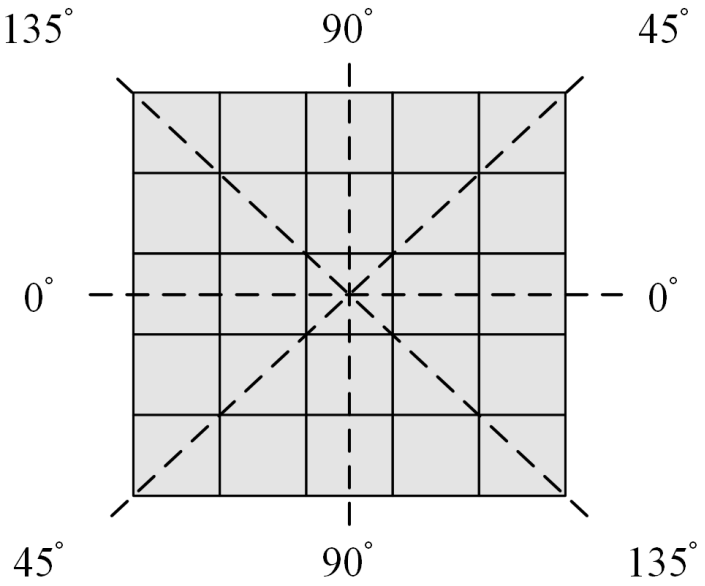


Figure 3.13. The configuration of spatial directions in $\tilde{\mathbf{u}}$. There are gray level runs in four directions: $0°, 45°, 90°$ and $135°$.

- Difference Entropy (DE)

$$\mathrm{DE} = -\sum_{\mathrm{j}\in\hat{\mathbb{U}}} \bar{\boldsymbol{\mathcal{B}}}_{i-j}(\mathrm{j}) \log_2 \bar{\boldsymbol{\mathcal{B}}}_{i-j}(\mathrm{j}), \tag{3.55}$$

  where $\mathrm{j} \in \hat{\mathbb{U}} = \{|i-j|, |i,j \in \mathbb{U}\}$.

A gray level run is defined over the image $\tilde{\mathbf{u}}$ to be a set of consecutive, collinear pixels having the same gray value (i.e. pixel intensity). The length of the run length is the number of pixels in the run [124]. The matrix element $(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})$ of the GLRL matrix $(\boldsymbol{\mathcal{H}})$ specifies the number of times that $\tilde{\mathbf{u}}$ contains a run of length $n_{\boldsymbol{\mathcal{H}}}$, in the given direction, consisting of pixels having gray value $u_{\boldsymbol{\mathcal{H}}}$ for $u_{\boldsymbol{\mathcal{H}}} \in \mathbb{U}$ and $n_{\boldsymbol{\mathcal{H}}} \in \mathbb{N}_{\boldsymbol{\mathcal{H}}}$. $N_{\boldsymbol{\mathcal{H}}} = |\mathbb{N}_{\boldsymbol{\mathcal{H}}}|$ is the number of different run lengths that are taken into account. There are four kinds of gray level runs with different spatial directions as shown in Fig. 3.13. Galloway proposed five features over GLRL,

- Short runs emphasis

$$\mathrm{RF}_1 = \frac{\sum_{u_{\boldsymbol{\mathcal{H}}}\in\mathbb{U}} \sum_{n_{\boldsymbol{\mathcal{H}}}\in\mathbb{N}_{\boldsymbol{\mathcal{H}}}} \frac{\boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})}{n_{\boldsymbol{\mathcal{H}}}^2}}{\sum_{u_{\boldsymbol{\mathcal{H}}}\in\mathbb{U}} \sum_{n_{\boldsymbol{\mathcal{H}}}\in\mathbb{N}_{\boldsymbol{\mathcal{H}}}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})}, \tag{3.56}$$

- Long runs emphasis

$$\mathrm{RF}_2 = \frac{\sum_{u_{\boldsymbol{\mathcal{H}}}\in\mathbb{U}} \sum_{n_{\boldsymbol{\mathcal{H}}}\in\mathbb{N}_{\boldsymbol{\mathcal{H}}}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}}) n_{\boldsymbol{\mathcal{H}}}^2}{\sum_{u_{\boldsymbol{\mathcal{H}}}\in\mathbb{U}} \sum_{n_{\boldsymbol{\mathcal{H}}}\in\mathbb{N}_{\boldsymbol{\mathcal{H}}}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})}, \tag{3.57}$$

- Gray level nonuniformity

$$\mathrm{RF}_3 = \frac{\sum_{u_{\boldsymbol{\mathcal{H}}} \in \mathbb{U}} \left(\sum_{n_{\boldsymbol{\mathcal{H}}} \in \mathbb{N}_{\boldsymbol{\mathcal{H}}}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}}) \right)^2}{\sum_{u_{\boldsymbol{\mathcal{H}}} \in \mathbb{U}} \sum_{n_{\boldsymbol{\mathcal{H}}} \in \mathbb{N}_{\boldsymbol{\mathcal{H}}}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})}, \tag{3.58}$$

- Run length nonuniformity

$$\mathrm{RF}_4 = \frac{\sum_{n_{\boldsymbol{\mathcal{H}}} \in \mathbb{N}_{\boldsymbol{\mathcal{H}}}} \left(\sum_{u_{\boldsymbol{\mathcal{H}}} \in \mathbb{U}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}}) \right)^2}{\sum_{u_{\boldsymbol{\mathcal{H}}} \in \mathbb{U}} \sum_{n_{\boldsymbol{\mathcal{H}}} \in \mathbb{N}_{\boldsymbol{\mathcal{H}}}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})}, \tag{3.59}$$

- Run percentage

$$\mathrm{RF}_5 = \frac{\sum_{n_{\boldsymbol{\mathcal{H}}} \in \mathbb{N}_{\boldsymbol{\mathcal{H}}}} \sum_{u_{\boldsymbol{\mathcal{H}}} \in \mathbb{U}} \boldsymbol{\mathcal{H}}(u_{\boldsymbol{\mathcal{H}}}, n_{\boldsymbol{\mathcal{H}}})}{N_x N_y}. \tag{3.60}$$

The implementation of texture features requires the discretization of pixel intensities. How the discretization is realized is important for the texture features. For instance, how many intervals are taken into account or whether nonlinear transform is demanded to emphasize the information of low intensity value pixels. In order to study the effect of discretization on the texture features, we adopt the following transforms in this thesis:

$$\tilde{u} = \mathrm{round}\left(\frac{u}{\mathrm{int}}\right), \tag{3.61}$$

$$\tilde{u} = \mathrm{round}\left(u^{\mathrm{index}}\right), \tag{3.62}$$

$$\tilde{u} = \mathrm{round}\left(\log_{\mathrm{base}}(1+u)\right), \tag{3.63}$$

where "round" is the operation of rounding the value to the nearest integer, u is the element of the array $\mathbf{u}$ (i.e. the observation in Section 2.2), $\tilde{u}$ is the element of the integer valued image $\tilde{\mathbf{u}}$, and int is a positive integer. The one in Equation (3.61) is a discretization with linear transform and another two discretization schemes with typical nonlinear transforms are given by Equations (3.62) and (3.63).

In Fig. 3.14, Fig. 3.15 and Fig. 3.16 there are examples to demonstrate the effect of the discretization with different transforms. The linear transform has little impact on the illustration of the image structures, cf. Fig. 3.14. In contrast, the nonlinear transform in Fig. 3.15 and Fig. 3.16 can emphasize some parts of the image structures depending on their parameter settings. In order to evaluate the influences of the discretization schemes with different parameters, a quantitative analysis to assess the resulting features is carried out on the basis of our database. We choose the MI (cf. Equation (4.1) in Chapter 4) of individual features for this assessment. A great value of MI indicates a high relevance of the associated feature. One discretization scheme with a certain parameter setting can provide us with a group of texture features.

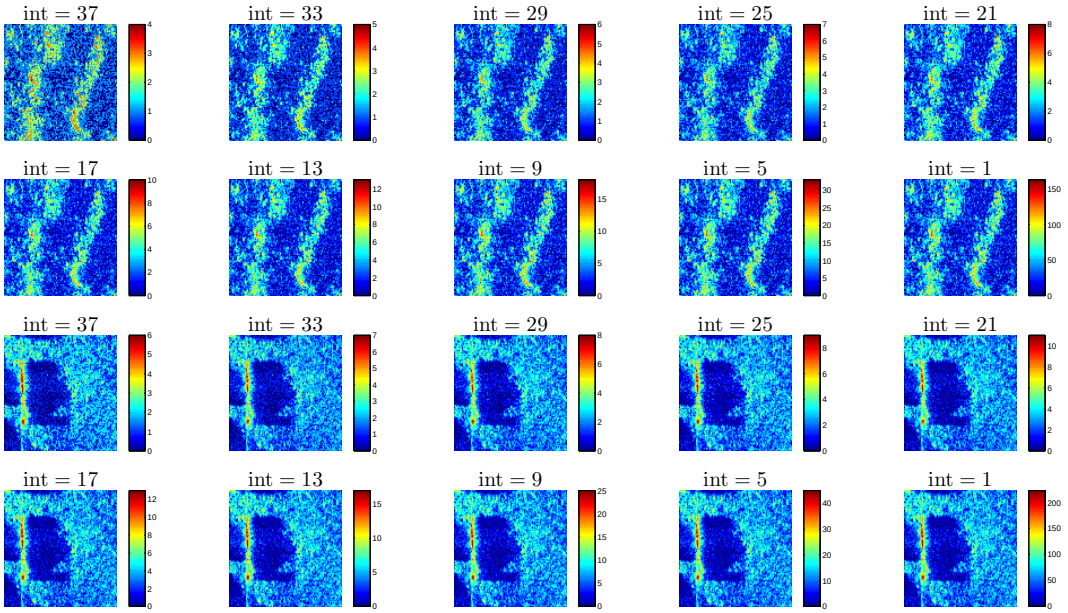


Figure 3.14. Linear discretization with different int as shown in Equation (3.61).

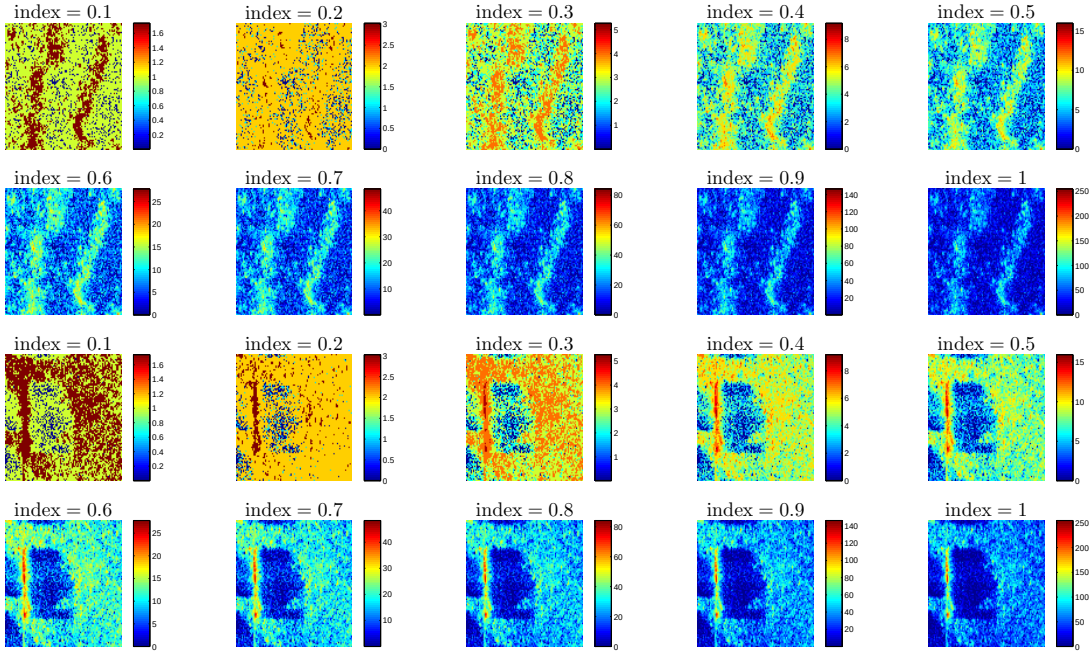


Figure 3.15. Nonlinear discretization with different power indices as shown in Equation (3.62).

We calculate the MIs of individual features, as well as the average of these MIs, in this group. In Fig. 3.17, we depict the curves denoting the averages of the MIs corresponding to the discretization schemes with different parameter settings.

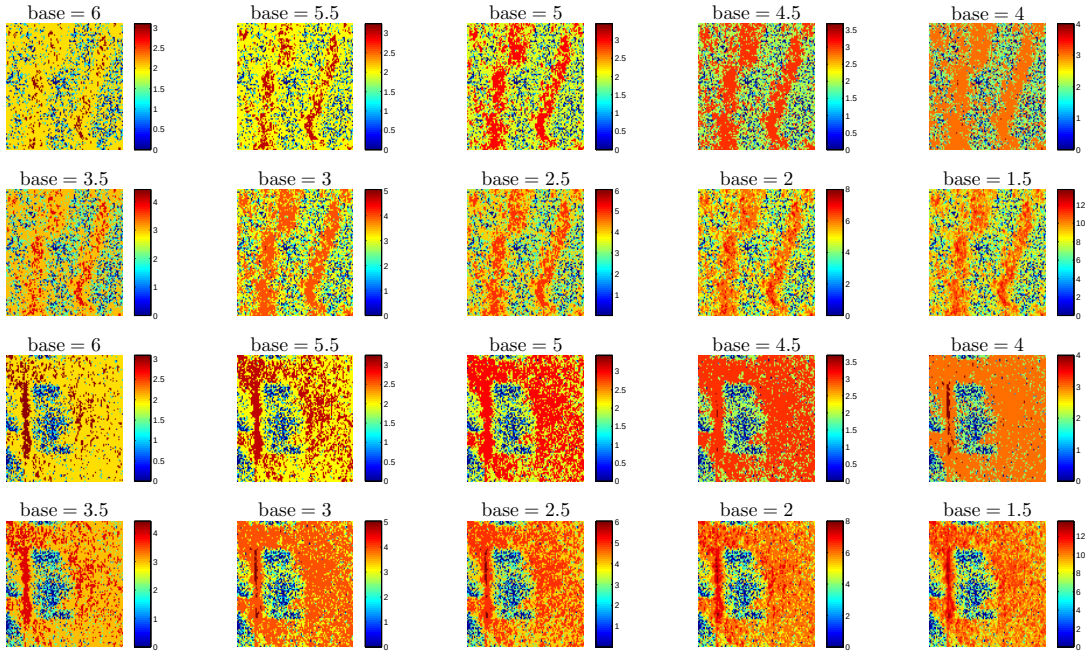


Figure 3.16. Nonlinear discretization with different logarithm bases as shown in Equation (3.63).

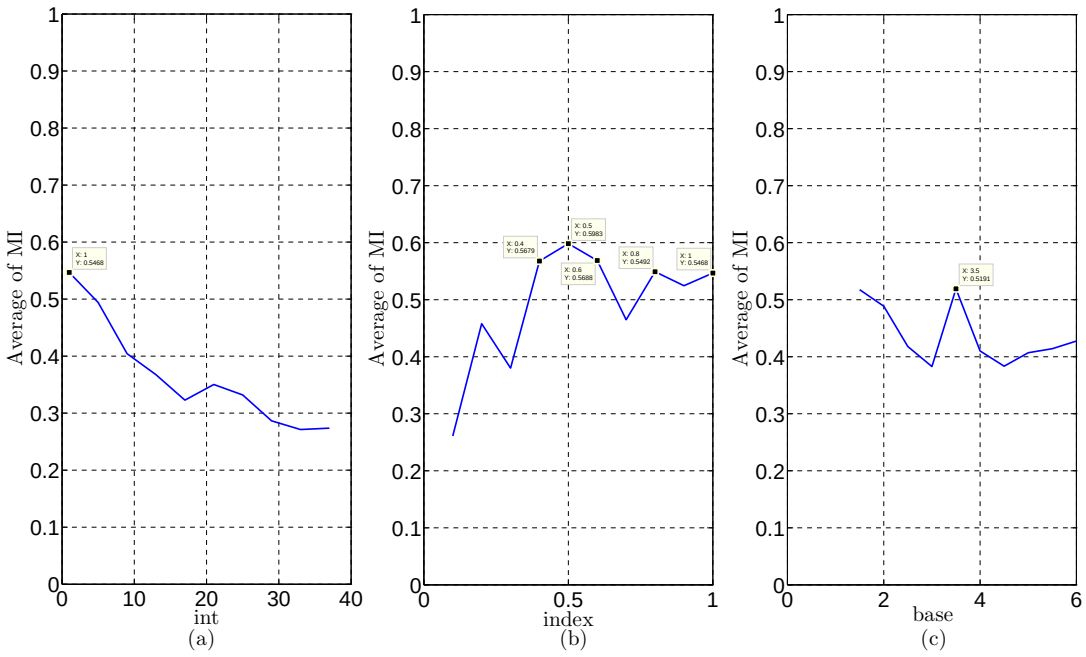


Figure 3.17. The MI averages of the texture features obtained by the discretization associated with different parameters. (a): The discretization using Equation (3.61). (b): The discretization using Equation (3.62). (c): The discretization using Equation (3.63).

The linear discretization is not able to improve the feature extraction. In contrast, the nonlinear transform can improve the texture feature extraction, but not always. The best performance is obtained when the discretization adopts the nonlinear discretization in Equation (3.62) with index = 0.5. Hence, images are discretized according to Equation (3.62) with index = 0.5 before being forwarded to texture feature calculation.

3.4 Conclusions

This chapter deals with the feature extraction. The features involved in the design of our automatic detection and automatic classification system have been introduced as well as their characteristics.

Even when the features are the key factors that have significant influence on the classification performance, few authors are willing to make the effort to describe the details about how their features are extracted as well as their associated motivations since it seems to be trivial. For the sake of clarity and completeness, we have gone through all of the features involved in the feature set in this chapter. Three types of features are considered, namely object region features, contour features and texture features. Besides the geometrical features that can be found in the literature, we proposed several new geometrical features that are suitable to our application. Their extraction methods, motivations and performances are described.

The discretization of the pixel intensity has a great influence on the texture features since the contrast of the image could be changed by using certain nonlinear transformation. This influence is quantitatively studied. In our application, the best one is the discretization scheme with nonlinear transform.

Chapter 4

Feature Selection Using a Novel Relevance Measure

A novel feature selection scheme is considered in this chapter. As mentioned in Chapter 3, the feature selection is conducted during the system design phase (cf. Fig. 1.2). Its results are used to guide the feature extraction in object classification phase to extract those relevant features. They are designated to prepare the inputs of the fourth step along the ADAC processing chain collaboratively. Without the knowledge about the relevant features for our application, the feature extraction is designed to include as many features as possible in the system design phase. The step of feature selection deals with the removing of unwanted features from the set **O** so that the danger of encountering the curse of dimensionality (cf. Fig. 1.3 in Chapter 1) can be avoided.

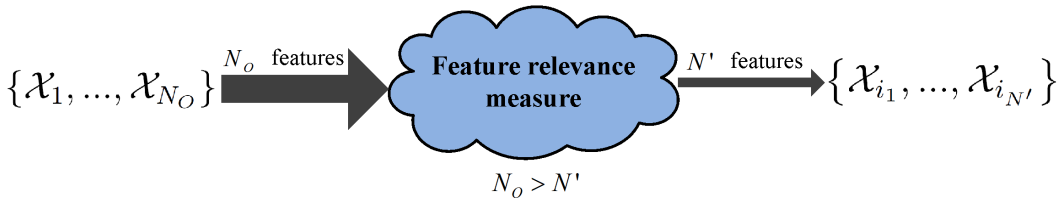


Figure 4.1. The filter method for feature selection. There are N_O features as input on the left side, where $\mathcal{X}_n$ with $1 \leq n \leq N_O$ denotes the n-th feature. We choose N' useful features out of the total number of N_O.

A widely adopted feature selection method that chooses the most relevant features out of the feature set is called the filter method [56] as shown in Fig. 4.1. Rather than taking the classification performance associated with specific classifiers as the selection criterion (i.e. wrapper methods), the filter method adopts a feature relevance measure to quantify the dependency of features on the types of objects. Mutual information (MI) has been widely applied as a relevance measure [125]. Despite its ability to catch arbitrary correlations between the features and the object types, not all of the captured information can be interpreted by classifiers, i.e. in reality, an arbitrary function cannot always be perfectly approximated by a learning machine. Bell *et al.* in [126] proposed a MI-based relevance measure to evaluate the additional classification-relevant information contributed by a candidate feature. Their relevance measure implicitly incorporated the idea of joint entropy minimization. It discarded the information which is irrelevant for classification regarding the training data. Features selected according to the relevance measure in [126] could be able to separate objects of different classes in

training data perfectly. However, the generalization to test data could encounter problems due to the dissimilarity between training data and test data which often occurs in practice. For instance, if a set of test data is dissimilar to training data because of the higher noise level in the test data, the features selected according to the relevance measure proposed by Bell *et al.* might not be adequate for this set of test data. Accordingly, the performance of classification could degrade. Brown *et al.* [127] reviewed three filter methods [128–130]. The relevance measures of these three methods consist of two parts. One is the MI measuring the classification-relevant information provided by a candidate feature, and the other is a redundancy part quantifying the duplicate information between this candidate feature and the features that are already selected. The relevance measures are constructed by extracting the redundancy part from the MI. The three methods are different in the way how they determine the redundancy part. In [128] and [130], firstly, the amounts of duplicate information (ADI) between the candidate feature and each already selected feature were computed and summed up. Then, Battiti *et al.* in [128] derived the redundancy part by multiplying the sum of ADI with a predefined factor, and Peng *et al.* in [130] built their redundancy part by dividing the sum of ADI by the number of selected features. In [127], the difference between these two redundancy parts was reviewed and interpreted. On the one hand, through a predefined factor, Battiti *et al.* implicitly quantified their belief in the assumption that features are pairwise class-conditionally independent. On the other hand, the redundancy part of Peng *et al.* inferred that a stronger belief was put in the assumption that features are pairwise independent as the size of selected features grows. Kwak *et al.* [129] improved the method in [128] by exploiting the additional assumption that the information is uniformly distributed in the calculation of ADI. However, the assumptions made in [128–130] do not generally hold in applications. Moreover, one has to manually set how many features to choose when employing these three methods. Alternatively, Kira *et al.* in [86] proposed a distance-based relevance measure, i.e. Relief weight. It was used to describe the extent of the distinction among different classes. As the Relief weight increases, there is less overlap between the classes. Accordingly, the features that have largest Relief weights are added to the feature subset. The redundancy among features was not specified and the setting of the threshold for those highest Relief weights was also *ad-hoc*.

Regarding to the limitation of the methods mentioned above, we propose a novel feature relevance measure called composite relevance measure in this chapter. It uses a novel feature relevance measure called the composite relevance measure (CRM), which combines the MI, Shannon entropy (SE) and the modified Relief weight (mRW). Both linear and nonlinear combinations are considered. The MI supervises the sufficiency of the selection. The consideration of SE in the CRM is crucial to avoid both overfitting

and underfitting. The Relief weight was originally proposed for binary-class problems to evaluate individual features. It is extended to be not only applicable to multi-class problems but also able to evaluate the relevance of the combinations of individual features, i.e. feature sets. The inclusion of mRW helps in making the captured information more manageable so that it can be learned by most of the classifiers.

The remainder of this chapter is organized as follows. Sec. 4.1 reviews the MI and conditional MI, and the mRW is presented in Sec. 4.2. The filter method using a novel feature relevance measure is introduced in Sec. 4.3. The numerical studies of the proposed filter method are carried out in Sec. 4.4.

4.1 Information based Relevance Measure

Let $\mathbf{S} \in \mathbf{O}$ be a selection of features, where $N_S = |\mathbf{S}|$ is the cardinality of the set $\mathbf{S}$. Let vector $\boldsymbol{\chi}_S^{(m)}$ be a point in the space $\mathbb{F}$ induced by $\mathbf{S}$ for $\dim(\mathbb{F}) = N_S$. The random variable (RV) C denotes the class index of the objects, and $c^{(m)} \in \boldsymbol{\mathcal{C}}$ is its m-th realization, where $\boldsymbol{\mathcal{C}} = \{c_1, c_2, ..., c_{N_c}\}$ contains all possible values of class indices.

The MI, which quantifies the information commonly found in two groups of RVs, e.g. between C and $\mathbf{S}$, is a suitable measure to specify the classification-relevant information contained in $\mathbf{S}$. It is defined as

$$I(\mathbf{S}, C) = H(C) - H(C|\mathbf{S}), \tag{4.1}$$

where the SE, $H(C)$, and the conditional SE, $H(C|\mathbf{S})$, are given as

$$H(C) = -\sum_{c \in \boldsymbol{\mathcal{C}}} p(c) \log p(c), \tag{4.2}$$

$$H(C|\mathbf{S}) = -\sum_{c \in \boldsymbol{\mathcal{C}}} \int_{\mathbb{F}} p(c, \boldsymbol{\chi}_S) \log p(c|\boldsymbol{\chi}_S) \mathrm{d}\boldsymbol{\chi}_S. \tag{4.3}$$

Moreover, the conditional mutual information (CMI) yields the net information that can be provided by the candidate feature $\mathcal{X}_{n'} \in \mathbf{O}' = \mathbf{O} \backslash \mathbf{S}$ when $\mathbf{S}$ is known. The $\mathbf{O} \backslash \mathbf{S}$ denotes set subtraction of $\mathbf{S}$ from $\mathbf{O}$, and the CMI is defined as

$$\begin{aligned} I(\mathcal{X}_{n'}, C|\mathbf{S}) &= H(C|\mathbf{S}) - H(C|\{\mathbf{S}, \mathcal{X}_{n'}\}) \\ &= I(\{\mathbf{S}, \mathcal{X}_{n'}\}, C) - I(\mathbf{S}, C). \end{aligned} \tag{4.4}$$

When the quantity $I(\mathcal{X}_{n'}, C|\mathbf{S})$ is large, it means that this candidate feature $\mathcal{X}_{n'}$ is still a relevant feature, even when the $\mathbf{S}$ is given. Thus, this measure is very useful when a

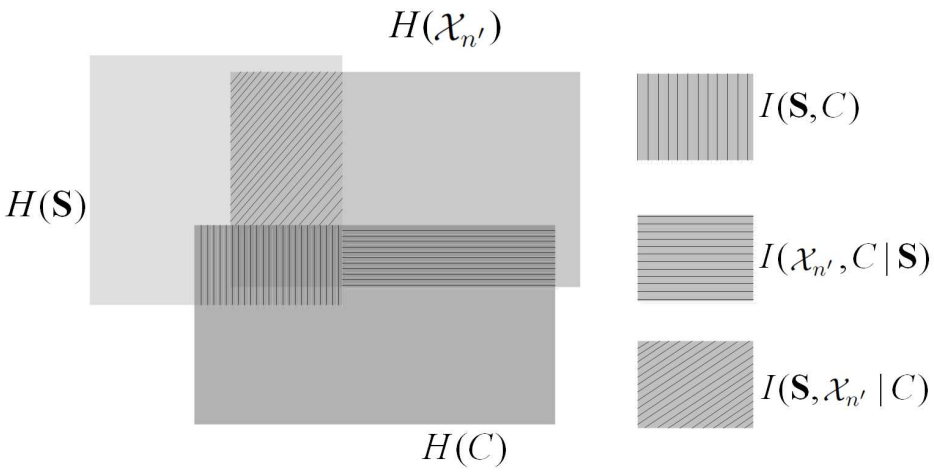


Figure 4.2. An illustration of MI and CMI. $I(\mathbf{S},C)$ denotes the classification-relevant information contained in $\mathbf{S}$, and $I(\mathcal{X}_{n'},C|\mathbf{S})$ is the additional classification-relevant information contributed by $\mathcal{X}_{n'}$. Moreover, $I(\mathbf{S},\mathcal{X}_{n'}|C)$ is the redundant information between $\mathcal{X}_{n'}$ and $\mathbf{S}$, which is classification irrelevant.

sequential forward procedure is applied. An illustration of MI and CMI is depicted in Fig. 4.2.

In the introduction of this chapter, methods like MIFS [128], MIFS-U [129] and mRMR [130] have been mentioned. The way they are used to assess the contribution of candidate features is going to be detailed here.

- MIFS: Mutual information based feature selection
  The criterion of MIFS is given as

$$J_{\mathrm{MIFS}}(\mathcal{X}_k) = I(\mathcal{X}_k,C) - \varkappa \sum_{\mathcal{X}_j\in\mathbf{S}} I(\mathcal{X}_k,\mathcal{X}_j), \tag{4.5}$$

  in which the belief in the assumption of pairwise class-conditional independence

$$p(\chi_k,\chi_j|c) = p(\chi_j|c)p(\chi_k|c), \tag{4.6}$$

  where χ_j,χ_k are the realizations of feature $\mathcal{X}_j$ and $\mathcal{X}_k$, is quantified by the factor $\varkappa\in[0,1]$.

- mRMR: Minimum redundancy maximum relevance feature selection
  The mRMR selects the features according to the measure

$$J_{\mathrm{mRMR}}(\mathcal{X}_k) = I(\mathcal{X}_k,C) - \frac{1}{|\mathbf{S}|}\sum_{\mathcal{X}_j\in\mathbf{S}} I(\mathcal{X}_k,\mathcal{X}_j), \tag{4.7}$$

in which the condition in Equation (4.6) is implicitly presumed to be valid and the belief in the pairwise independence between features

$$p(\chi_k, \chi_j) = p(\chi_k)p(\chi_j) \tag{4.8}$$

is controlled by the size of selection $\mathbf{S}$. When N_S approaches infinity, the features are believed to be totally pairwise independent.

- MIFS-U: Mutual information variable selection under uniform information distribution
  In addition to MIFS, it assumes that the information is distributed uniformly throughout the region of $H(\mathcal{X}_j)$ for $\mathcal{X}_j \in \mathbf{S}$. It evaluates the contribution of candidate features by using

$$J_{\text{MIFS-U}}(\mathcal{X}_k) = I(\mathcal{X}_k, C) - \varkappa \sum_{\mathcal{X}_j \in \mathbf{S}} \frac{I(\mathcal{X}_j, C)}{H(\mathcal{X}_j)} I(\mathcal{X}_k, \mathcal{X}_j), \tag{4.9}$$

  where the $\varkappa$ has a similar meaning as the one in Equation (4.5).

However, due to the limitation of their assumptions, they are not adequate to precisely estimate the real contribution of the candidate features. Moreover, what these measures are dealing with are the individual features. As mentioned in Chapter 3, the combination of individually *insignificant* features is possible to create a very relevant feature set, cf. Fig. 3.1. In that figure, the MIs of the example features $\mathcal{X}_1$ and $\mathcal{X}_2$ are $I(\mathcal{X}_1, C) = 0$ and $I(\mathcal{X}_2, C) = 0$, respectively. After combining both of them, we have $I(\{\mathcal{X}_1, \mathcal{X}_2\}, C) = 1$. This example indicates that the combination of $\mathcal{X}_1$ and $\mathcal{X}_2$ can provide more information together than by the sum of their parts, i.e. it is possible to have the following inequality:

$$I(\{\mathcal{X}_1, \mathcal{X}_2\}, C) > I(\mathcal{X}_1, C) + I(\mathcal{X}_2, C). \tag{4.10}$$

Although the measure of RELFSS [126]

$$J_{\text{RELFSS}}(\mathcal{X}_k) = \frac{I(\{\mathbf{S}, \mathcal{X}_k\}, C)}{H(\{\mathbf{S}, \mathcal{X}_k\})} \tag{4.11}$$

handles the combination of candidate feature and selected features, it implicitly incorporates the empirical theorem of *minimization of joint entropy*. This empirical theorem is not necessarily valid for all practical applications, for instance the underwater target recognition, cf. the performance analysis in [131].

4.2 The Modified Relief Weight

We proposed in [132] the novel distance measure

$$d\left(\boldsymbol{\chi}_S^{(m_1)}, \boldsymbol{\chi}_S^{(m_2)}\right) = d_{\mathrm{M}}\left(\boldsymbol{\chi}_S^{(m_1)}, \boldsymbol{\chi}_S^{(m_2)}\right) \exp\left(-\frac{d_{\mathrm{M}}\left(\boldsymbol{\chi}_S^{(m_1)}, \boldsymbol{\chi}_S^{(m_2)}\right)}{\mathrm{dist}_{\max}}\right), \tag{4.12}$$

where d_{M} represents the Manhattan distance (MD) between the two input vectors, and $\mathrm{dist}_{\max}$ is the maximum distance, which is given by

$$\mathrm{dist}_{\max} = \max_{m_1, m_2 \in \{1,\ldots,M\}} d_{\mathrm{M}}\left(\boldsymbol{\chi}_S^{(m_1)}, \boldsymbol{\chi}_S^{(m_2)}\right). \tag{4.13}$$

It is assumed that if an object has a relatively large distance to its nearest neighbors, it could be considered as an outlier. The distance information obtained from this object is no longer plausible. Hence, we discount the distance in Equation (4.12) by an MD driven factor, and its curve is depicted in Fig. 4.3. The non-decreasing curve shows that

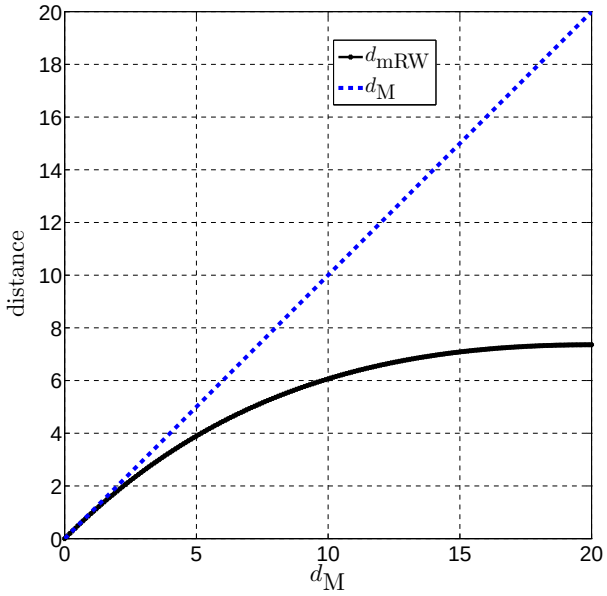


Figure 4.3. The curve of the proposed novel distance in Equation (4.12). The $\mathrm{dist}_{\max} = 20$ in this case. The dashed line denotes a distance without the correction by $\exp\left(-\frac{d_{\mathrm{M}}\left(\boldsymbol{\chi}_S^{(m_1)}, \boldsymbol{\chi}_S^{(m_2)}\right)}{\mathrm{dist}_{\max}}\right)$.

the d_{mRW} stops increasing itself when d_{M} approaches the maximum MD value $\mathrm{dist}_{\max}$ (in the case shown in Fig. 4.3 $\mathrm{dist}_{\max} = 20$). According to the proposed distance, we find two neighbors in the neighborhood of $\boldsymbol{\chi}_S^{(m)}$; one is the nearest neighbor $\boldsymbol{\chi}_S^{(\mathrm{hit},m)}$ in

the same class of $\boldsymbol{\chi}_S^{(m)}$, and the other is the nearest neighbor $\boldsymbol{\chi}_S^{(\mathrm{mis},m)}$ out of the other classes, which are different from the one of $\boldsymbol{\chi}_S^{(m)}$. Then, the mRW assigned to the set $\mathbf{S}$ is

$$W(\mathbf{S}) = \frac{1}{M}\sum_{m=1}^{M} w(m), \tag{4.14}$$

where $w(m)$ is given by

$$w(m) = d\left(\boldsymbol{\chi}_S^{(m)}, \boldsymbol{\chi}_S^{(\mathrm{mis},m)}\right) - d\left(\boldsymbol{\chi}_S^{(m)}, \boldsymbol{\chi}_S^{(\mathrm{hit},m)}\right). \tag{4.15}$$

According to the discussion above, the proposed distances of an outlier to its $\boldsymbol{\chi}_S^{(\mathrm{mis},m)}$ and $\boldsymbol{\chi}_S^{(\mathrm{hit},m)}$ will be close, since their MDs to the nearest neighbors are close to $\mathrm{dist}_{\max}$; cf. Fig. 4.4. Due to this behavior, the $w(m)$ of an outlier tends to zero. It means that outliers have little influence on the value of $W(\mathbf{S})$, i.e. their information is suppressed. A huge mRW value indicates that the feature vectors for objects belonging to different classes are well separated. Hence, when the mRW of $\mathbf{S}$ is large, it means that the features in $\mathbf{S}$ are relevant.

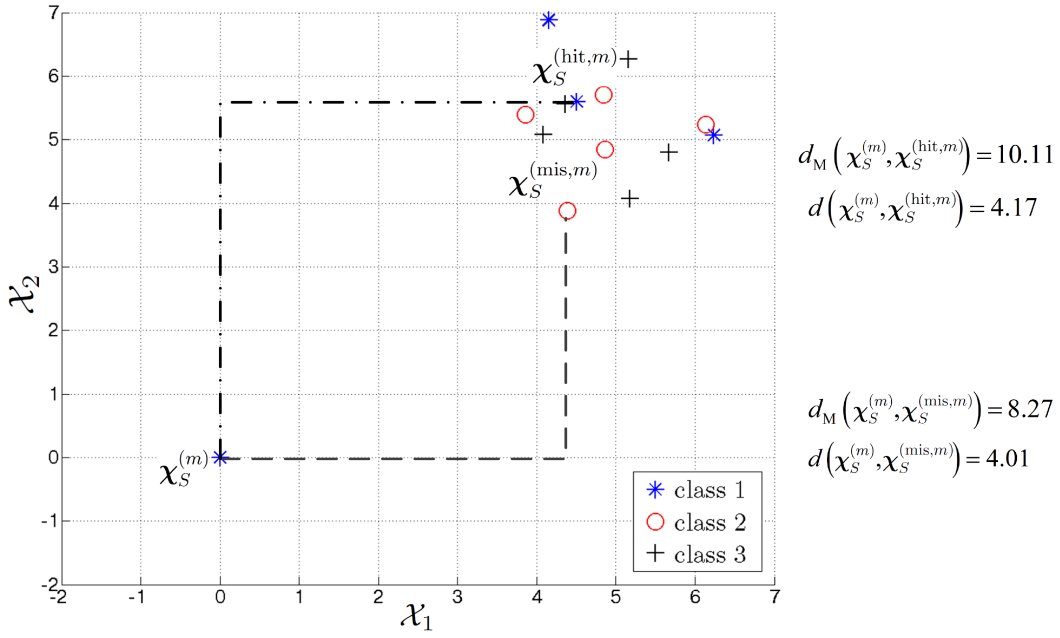


Figure 4.4. An example to illustrate the modified Relief weights (mRW) using features $\mathcal{X}_1$ and $\mathcal{X}_2$. The feature vector $\boldsymbol{\chi}_S^{(m)}$ for object m is an outlier. The Manhattan distances are depicted. The values of d_{M} and d are presented on the right side. It can be seen that the difference between $d\left(\boldsymbol{\chi}_S^{(m)}, \boldsymbol{\chi}_S^{(\mathrm{hit},m)}\right)$ and $d\left(\boldsymbol{\chi}_S^{(m)}, \boldsymbol{\chi}_S^{(\mathrm{mis},m)}\right)$ is much smaller than the one between $d_{\mathrm{M}}\left(\boldsymbol{\chi}_S^{(m)}, \boldsymbol{\chi}_S^{(\mathrm{hit},m)}\right)$ and $d_{\mathrm{M}}\left(\boldsymbol{\chi}_S^{(m)}, \boldsymbol{\chi}_S^{(\mathrm{mis},m)}\right)$.

The values of different features can cover very different ranges because they can have different physical meanings. Therefore, the mRWs obtained from different feature sets

cannot be compared fairly. A scaling of the features is required. We normalize the features against their standard deviation before forwarding them to the mRW evaluation. In addition, the mRW should be invariant concerning the number of objects, M. Therefore, the factor $\frac{1}{M}$ in Equation (4.14) is indispensable.

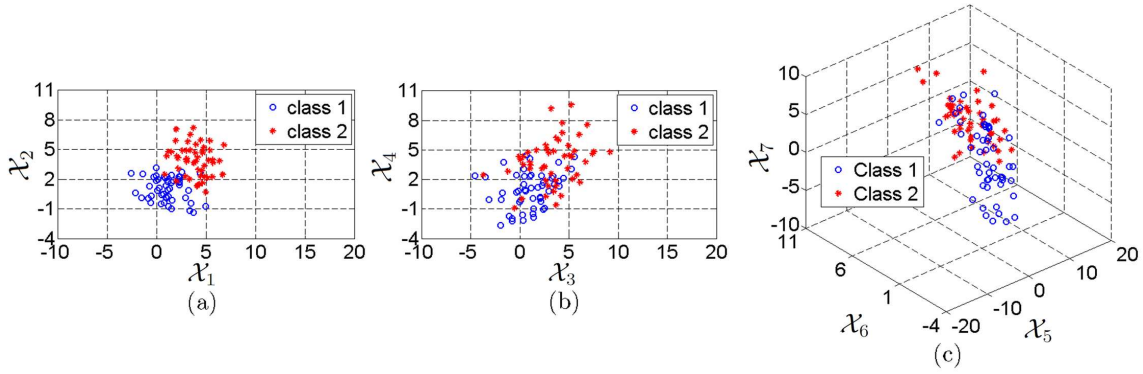


Figure 4.5. Objects are represented by feature vectors in different feature spaces. There are seven features in $\mathbf{O}$, out of which three feature selections are built, namely $\mathbf{S}_1 = \{\mathcal{X}_1, \mathcal{X}_2\}$, $\mathbf{S}_2 = \{\mathcal{X}_3, \mathcal{X}_4\}$ and $\mathbf{S}_3 = \{\mathcal{X}_5, \mathcal{X}_6, \mathcal{X}_7\}$. The feature vectors depicted in (a), (b) and (c) are induced by the sets $\mathbf{S}_1$, $\mathbf{S}_2$ and $\mathbf{S}_3$, respectively.

There is an example presented in Fig. 4.5 to clarify the properties of the mRW. The set $\mathbf{O}$ contains seven features and we build three feature selections with the features out of $\mathbf{O}$. For simplicity, they are set as $\mathbf{S}_1 = \{\mathcal{X}_1, \mathcal{X}_2\}$, $\mathbf{S}_2 = \{\mathcal{X}_3, \mathcal{X}_4\}$ and $\mathbf{S}_3 = \{\mathcal{X}_5, \mathcal{X}_6, \mathcal{X}_7\}$. As a result, we get $W(\mathbf{S}_1) = 0.1989$, $W(\mathbf{S}_2) = 0.0957$ and $W(\mathbf{S}_3) = 0.3063$. It is observed that the extent of the overlap between classes in Fig. 4.5(b) is much greater than that in Fig. 4.5(a). Accordingly, we have $W(\mathbf{S}_2) < W(\mathbf{S}_1)$. However, while considering $\mathbf{S}_1$ and $\mathbf{S}_3$, we find that although the extent of the overlap in Fig. 4.5(c) is also greater than the one in Fig. 4.5(a), the $W(\mathbf{S}_3)$ is still greater than the $W(\mathbf{S}_1)$. This could be attributed to the additional spatial dimension contributed by the third feature of $\mathbf{S}_3$. Thus, it is unreasonable to compare the mRWs obtained in spaces of different dimensionalities.

4.3 Maximum Composite Relevance Using a Sequential Forward Search Scheme

In general, the selection process of features can be denoted as a function such that we have

$$\mathbf{S} = \mathcal{T}(\mathbf{O}), \qquad \text{for} \quad \mathbf{S} \subseteq \mathbf{O}, \tag{4.16}$$

where $\mathcal{T}$ is the function used to select features. According to the data-processing inequality [133], we have $I(\mathcal{T}(\mathbf{O}), C) \leq I(\mathbf{O}, C)$, i.e. $I(\mathbf{S}, C) \leq I(\mathbf{O}, C)$. There is a possibility that the equality holds if C is independent of $\mathbf{O}$ conditioned on $\mathbf{S}$ as follows,

$$I(\mathbf{O}, C|\mathbf{S}) = 0. \tag{4.17}$$

If a feature selection $\mathbf{S}$ can fulfill the condition in Equation (4.17), it is denoted as a sufficient feature set (sFS). Obviously, if $\mathbf{S} = \mathbf{O}$, this feature selection is an sFS. The chain rule of SE is

$$H(\{\mathcal{X}_{n_1}, \mathcal{X}_{n_2}, ..., \mathcal{X}_{n_i}\}) = \sum_{j=1}^{i} H\left(\mathcal{X}_{n_j} | \left\{\mathcal{X}_{n_{j-1}}, ..., \mathcal{X}_{n_1}\right\}\right), \tag{4.18}$$

so that the $H(\mathbf{S})$ is a non-decreasing function of the feature number in $\mathbf{S}$. If there is a feature $\mathcal{X}_n \in \mathbf{O}$ with $H(\mathcal{X}_n|\mathbf{O}\backslash\mathcal{X}_n) = 0$, this $H(\mathbf{S})$ is able to achieve its saturation before $\mathbf{S} = \mathbf{O}$. We apply the mutual information toolbox, which has been developed by Brown *et al.* according to the methods presented in [127], to our database (cf. Sec. 4.4.1), and estimate the $H(\mathbf{S})$ and $I(\mathbf{S}, C)$ of the selections with increasing N_S. The results are illustrated in Fig. 4.6. The redundancy between features is high enough so that both $H(\mathbf{S})$ and $I(\mathbf{S}, C)$ can reach their saturation before N_S reaches $|\mathbf{O}|$.

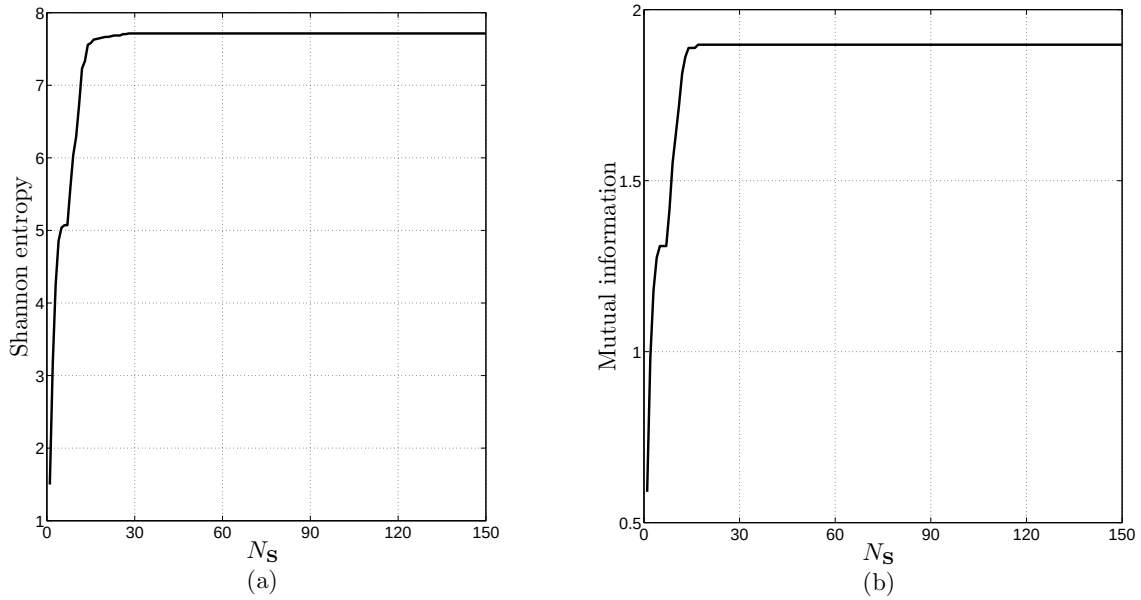


Figure 4.6. The curve of MI and SE with increasing number of features according to their sequence in the database, (a) Shannon entropy and (b) mutual information. No operation is made to rank the features regarding their relevance. The sequence of the features is subject to their extraction. The x-axis is number of the features that are taken into consideration.

Bell *et al.* in [126] pointed out that an sFS may not be unique. Consequently, an exclusive consideration of MI is inadequate. As described in Sec. 4.2, the mRW evaluates the feature relevance in an alternative way, in which the relevance is quantified by a distance measure rather than the information entropy. The consideration of the mRW could help us in choosing an optimal $\mathbf{S}$ among the sFS's. However, the mRW provides nothing about the data complexity, which is very important for avoiding underfitting and overfitting. Thus, the inclusion of $H(\mathbf{S})$ is also necessary. Fei *et al.* in [131] demonstrated that feature subsets with larger $H(\mathbf{S})$ are more likely to provide better results. The joint consideration of MI, SE and mRW in the CRM can be realized through either the weighted arithmetic average (J_{a}) or the weighted geometric average (J_{g}) as follows:

$$J_{\mathrm{a}}(\mathbf{S}) = (1-\gamma_{a,W}-\gamma_{a,H})I(\mathbf{S},C)+\gamma_{a,W}W(\mathbf{S})+\gamma_{a,H}H(\mathbf{S}), \tag{4.19}$$

$$J_{\mathrm{g}}(\mathbf{S}) = I(\mathbf{S},C)^{(1-\gamma_{g,W}-\gamma_{g,H})}W(\mathbf{S})^{\gamma_{g,W}}H(\mathbf{S})^{\gamma_{g,H}}, \tag{4.20}$$

where $0<\gamma_{a,W},\gamma_{a,H}<1, 0<\gamma_{a,W}+\gamma_{a,H}<1$ and $0<\gamma_{g,W},\gamma_{g,H}<1, 0<\gamma_{g,W}+\gamma_{g,H}<1$. A comprehensive assessment of the feature relevance is now available with the help of the CRMs in Equations (4.19) and (4.20). Nevertheless, there is still a difficulty in monitoring whether there are sufficient features selected in $\mathbf{S}$, since the CRM contains the distance measure mRW that does not provide any information about the amount of the classification-relevant information contained in $\mathbf{S}$. Moreover, the discussion in Sec. 4.2 has already shown that the comparison between mRWs associated with different N_S is unreasonable. A higher dimensional feature vector can increase the scale of the distance. In consequence, another measure excluding the consideration of mRW is required to form a stopping rule. It is called the sufficiency of $\mathbf{S}$. The sufficiency associated with the CRM is defined by

$$G(\mathbf{S}) = \max\left\{I(\mathbf{O},C)-I(\mathbf{S},C), H(\mathbf{O})-H(\mathbf{S})\right\}. \tag{4.21}$$

When there are enough features selected in $\mathbf{S}$, the $G(\mathbf{S})$ converges to zero. Evidently, the $\mathbf{S}$ selected according to this sufficiency is an sFS. So far, our task of finding the optimal features is converted to the maximization of the CRM subject to the convergence of $G(\mathbf{S})$ to zero. The complete search space is the set of all possible combinations of N_S features out of N for $1 \leq N_S \leq N$ leading to an NP-hard problem. The most commonly adopted sequential forward search (SFS) scheme is chosen to bypass this difficulty. In SFS, the $I(\mathbf{S},C)$ is fixed for each iteration loop, and the CMI in Equation (4.4) depends only on $I(\{\mathbf{S},\mathcal{X}_{n'}\},C)$. Thus, the $I(\{\mathcal{X}_{n'},\mathbf{S}\})$ is calculated with the help of the implementation provided by Brown *et al.* in [127]. The proposed feature selection algorithm called maximum composite relevance measure using a sequential forward search scheme (MCRM-SFS) is depicted in Fig. 4.7. The MCRM-SFS employing the $J_{\mathrm{a}}(\mathbf{S})$ is denoted as MCRM-SFSA and the one using $J_{\mathrm{g}}(\mathbf{S})$ is MCRM-SFSG.

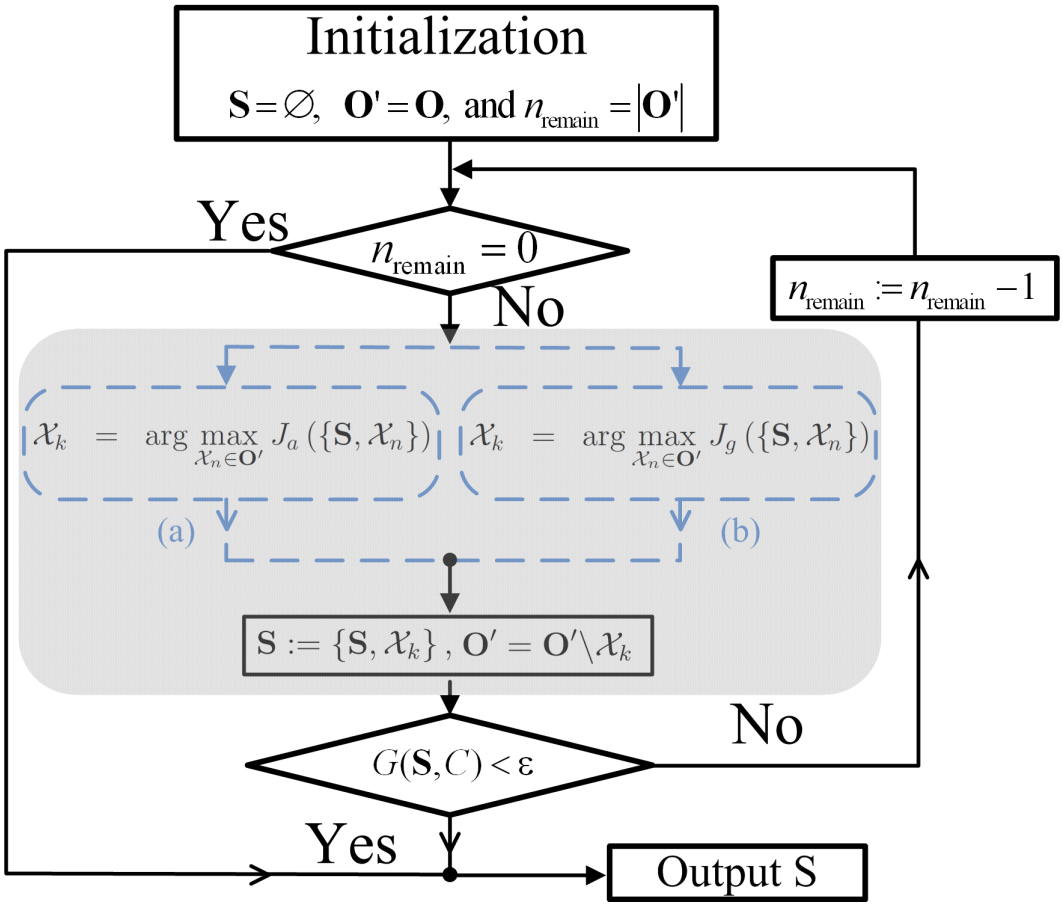
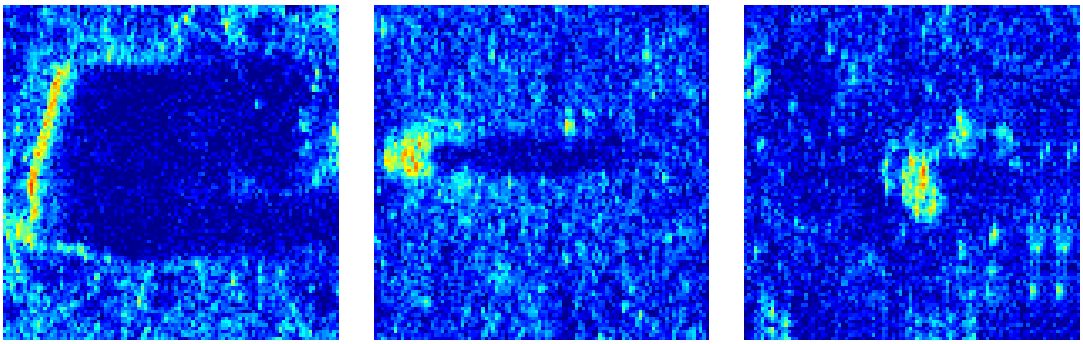


Figure 4.7. The flow chart of the MCRM-SFS. In the shadow block, one chooses either the left dashed path or the right dashed path. (a): When J_a is chosen, it is MCRM-SFSA. (b): When J_a is chosen, it is MCRM-SFSG.

4.4 The Numerical Studies of MCRM-SFS

4.4.1 Database Description

The database for numerical tests is provided by ATLAS ELEKTRONIK GmbH Bremen. There are in total $M = 210$ MLOs in this database, which includes 67 cylinder mines, 118 truncated cone mines and 45 rocks; cf. Fig. 4.8. The feature set contains the geometrical features of the MLOs and the texture features of the ROI as described in Chapter 3. Considering both the highlights and the shadows, there are 56 geometrical features (cf. Table 3.2). Moreover, we take the COOC matrix and GLRL matrix to describe the textures. Due to a lack of *a priori* knowledge about parameter settings providing significant features, we allowed several settings simultaneously. The setting of COOC depends on the offset between pixels, i.e. the absolute value of d_x and d_y,



(a) (b) (c)

Figure 4.8. Examples of the objects in our database: (a) a cylinder mine, (b) a truncated cone mine and (c) a rock.

and their spatial relationship, i.e. $0°, 45°, 90°$ or $135°$ (cf. Fig. 3.12). All the four spatial relationships are taken into account and the offsets are chosen as $1, 2, 3, 5$ and 10. Accordingly, there are $4 \times 5 = 20$ COOC matrices with different settings, and each of them can induce 12 features. Therefore, the number of COOC matrix based features is 240. The GLRL matrix also relies on parameters such as the maximum considered run length and the spatial directions. We consider the four spatial directions in Fig. 3.13. The maximum considered run length could be $10, 30$ and 50. Then, there are $3 \times 4 = 12$ GLRL matrices associated with different settings. Every GLRL matrix can induce five features. Thus, the number of the GLRL matrix based features equals 60. Finally, there are $N_O = 356$ features in the set $\mathbf{O}$. All the features are normalized against their standard deviation.

4.4.2 Classifiers Applied in Tests

Four classifiers are implemented for the numerical assessment, i.e. PNN is the probabilistic neural network [60], KNN is the k-nearest neighbor algorithm, and KNND [63] is the KNN assisted by Dempster-Shafer evidence theory, SVMG denotes the support vector machine (SVM) [134] using a Gaussian kernel. Let set

$$\mathfrak{E} = \{\text{KNN, KNND, PNN, SVMG}\} \tag{4.22}$$

be the set of the implemented classifiers. The features selected by MCRM-SFS are fed to those classifiers. For the implementation of the SVMG, the toolbox created by Canu *et al.* [135] is used. The width of the Gaussian kernel is set to 3, which is an empirically optimal setting for our database. Since the data is not perfectly separable,

we set the margin penalty equal to a moderate value of 1000. As for the KNN and the KNND, the number of neighbors taken into consideration is an important parameter for the classification. In our studies, it is found that satisfactory results are mostly achieved when seven neighbors are considered. In KNND, Euclidean distances are converted to belief values, which denote the support provided to hypotheses regarding the classification of objects. The KNND makes classification according to the total belief assigned to individual hypotheses as detailed in [63]. The PNN described by Duda *et al.* in [65] is employed. It is a three-layer neural network, i.e. consisting of the input layer, the pattern layer and the category layer. Each unit of the input layer is connected to all the units in the pattern layer. Each unit in the pattern layer, in turn, is connected to one unit in the category layer. The free parameter associated with the nonlinear function involved in PNN is set to 0.4.

4.4.3 Numerical Tests

Let $\mathbf{\Gamma}_a = (\gamma_{a,W}, \gamma_{a,H})^T$ denote the parameter setting vector associated with MCRM-SFSA and let $\mathbf{\Gamma}_g = (\gamma_{g,W}, \gamma_{g,H})^T$ denote the one corresponding to MCRM-SFSG. A setting of $\mathbf{\Gamma}_a$ corresponds to a feature selection obtained by MCRM-SFSA and, similarly, a setting of $\mathbf{\Gamma}_g$ is associated with a feature selection given by MCRM-SFSG. In order to find the optimal settings for MCRM-SFSA and MCRM-SFSG, we vary the settings of $\mathbf{\Gamma}_a$ and $\mathbf{\Gamma}_g$ to obtain multiple feature selections. A feature selection out of them is chosen, and feature vectors are calculated according to this feature selection. Then, these feature vectors are used as inputs of the classifiers in Equation (4.22). The accuracy of the classification based on this feature selection can be evaluated, and the performance associated with the corresponding parameter setting (i.e. $\mathbf{\Gamma}_a$ or $\mathbf{\Gamma}_g$) can be assessed as well. Hence, the search for optimal parameter settings for MCRM-SFSA and MCRM-SFSG becomes possible. Since the number of objects in the database is limited, a leave-one-out scheme is used in the numerical studies. Classifiers are trained on a set that includes $M-1$ objects out of the database. The test set contains the single remaining object, on which the classification test is carried out. In order to test through all the objects in the database, this leave-one-out scheme is repeated M times. Thus, every object in the database has been used as the test object once. Then, each object has an associated classification result. Hence, comparing these results with the ground truth, the performance of the proposed filter method can be evaluated by considering the classification accuracy, which is quantified by the empirical classification rate

$$\rho = \frac{m_{\text{correct}}}{M}, \tag{4.23}$$

where m_{correct} is the number of objects whose classification results are correct with regard to the ground truth. When classifier $\mathfrak{e} \in \mathfrak{E}$ is used, the classification rates of the MCRM-SFSA and the MCRM-SFSG are denoted as $\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ and $\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$, respectively. They are depicted in Fig. 4.9 and Fig. 4.10. The cases associated with $\gamma_{a,W} + \gamma_{a,H} \geq 1$ and $\gamma_{g,W} + \gamma_{g,H} \geq 1$ are set to zeros in the figures.

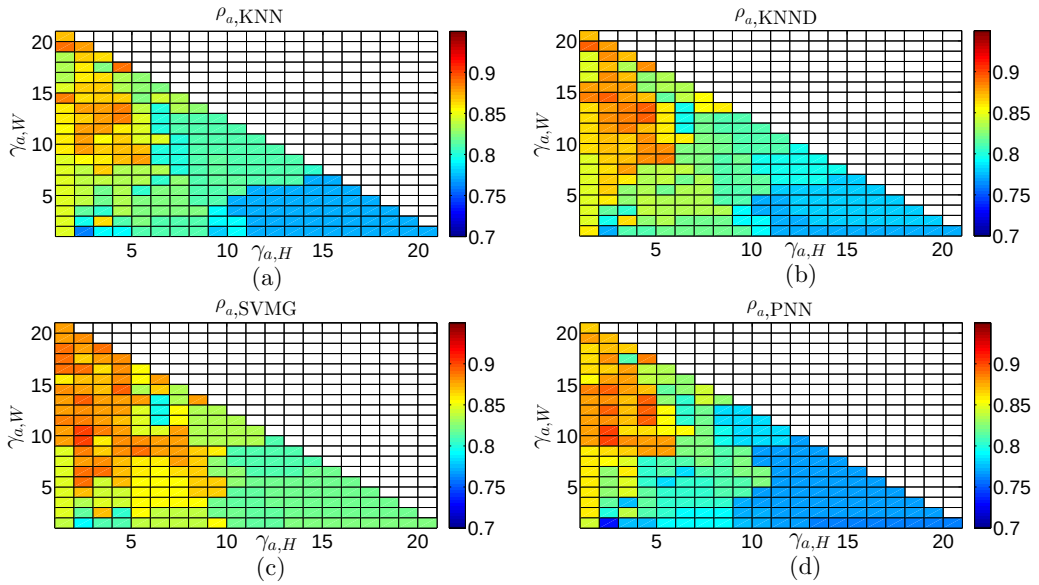


Figure 4.9. The $\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ for $\mathfrak{e} \in \mathfrak{E}$ corresponding to the features selected by MCRM-SFSA. The x- and y-axes denote the $\gamma_{a,H}$ and $\gamma_{a,W}$, respectively. (a) The results obtained by KNN, (b) by KNND, (c) by SVMG, (d) by PNN.

Analyzing the results given in Fig. 4.9 and Fig. 4.10, three facts are revealed. First of all, the differences of classification rates among different classifiers are not significant. Secondly, for the MCRM-SFSA, increasing the $\gamma_{a,W}$ can improve the classification results, which indicates the importance of the modified relief weight. Finally, for the MCRM-SFSG, the classification results do not change significantly as long as $\gamma_{g,W} > 0$ is large enough.

For the comparison, the methods mentioned in the introduction, i.e. RELFSS [126], mRMR [130], MIFS [128] and MIFS-U [129], have been implemented as reference. The classification rates using the features selected by RELFSS and mRMR are presented in Table 4.1, and the classification rates corresponding to MIFS and MIFS-U are given in Table 4.2 and Table 4.3, respectively. Apparently, the performance of RELFSS is worse than those of mRMR, MIFS and MIFS-U.

The implementation of RELFSS does not need a manual setting of the cardinality N_S. In contrast, the manual setting of N_S is demanded for the methods mRMR, MIFS

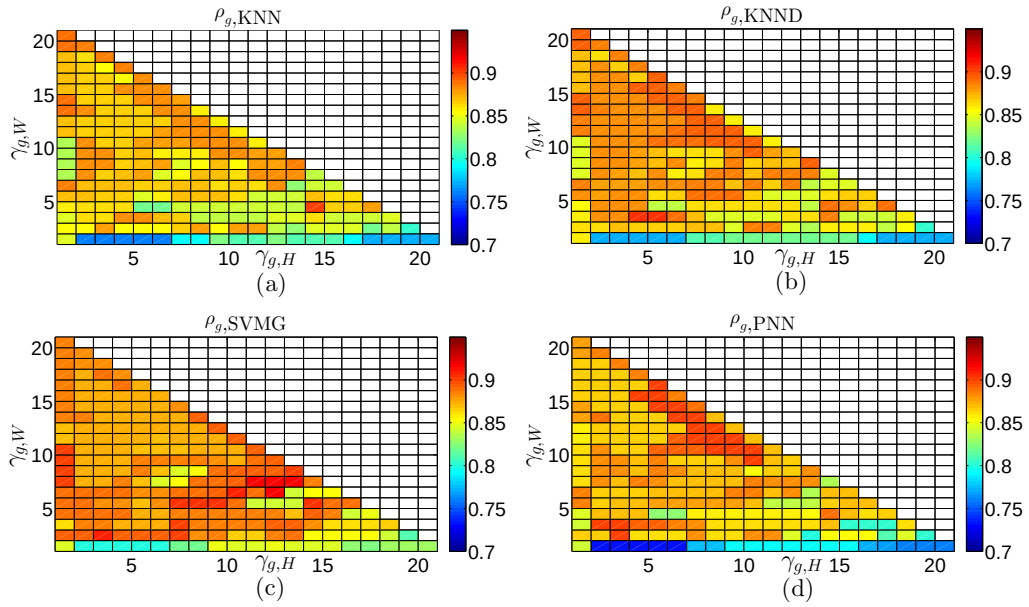


Figure 4.10. The $\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$ for $\mathfrak{e} \in \mathfrak{E}$ corresponding to the features selected by MCRM-SFSG. The x- and y-axes denote the $\gamma_{g,H}$ and $\gamma_{g,W}$, respectively. (a) The results obtained by KNN, (b) by KNND, (c) by SVMG, (d) by PNN.

Method	KNN	KNND	SVMG	PNN
RELFSS	0.7952	0.8000	0.8571	0.8143
mRMR	0.8667(5)	0.8762(8)	0.8810(9)	0.8619(5)

Table 4.1. The classification rates of various classifiers based on the selection methods RELFSS and mRMR. For mRMR, the associated optimal N_S-values are recorded in brackets.

$\varkappa$	KNN	KNND	SVMG	PNN
0	**0.8667**(11)	**0.8762**(11)	0.8810(9)	0.8619(9)
0.3	**0.8667**(7)	0.8714(7)	**0.8857**(14)	0.8667(9)
0.5	0.8476(8)	0.8714(11)	0.8952(11)	**0.8810**(5)
0.7	0.8381(7)	0.8524(7)	0.8714(12)	0.8429(3)
1	0.8286(8)	0.8286(8)	0.8571(13)	0.8524(10)

Table 4.2. The classification rates of various classifiers based on the selection method MIFS. The associated optimal N_S-values are recorded in brackets. The best results in individual columns are highlighted in bold.

$\varkappa$	KNN	KNND	SVMG	PNN
0	**0.8667**(11)	**0.8762**(11)	0.8810(9)	0.8619(9)
0.3	**0.8667**(7)	0.8714(7)	0.8857(5)	0.8714(16)
0.5	0.8476(8)	0.8714(11)	0.8714(3)	0.8571(16)
0.7	0.8381(7)	0.8524(7)	**0.8952**(12)	**0.8762**(5)
1	0.8286(8)	0.8286(8)	0.8762(8)	0.8714(7)

Table 4.3. The classification rates of various classifiers based on the selection method MIFS-U. The associated optimal N_S-values are recorded in brackets. The best results in individual columns are highlighted in bold.

and MIFS-U. The authors in [130] suggested probing with several possible values of N_S and employing the one with the best classification rate. It is found in our study that the cardinality N_S, which is greater than 20, can cause dramatic performance degradation for the classification using our database. Therefore, we vary N_S from 1 to 20. Each N_S is associated with a feature selection, i.e. a candidate. Feature vectors of objects are calculated based on this candidate, and subsequently used as inputs of a classifier. Then, the classification rate corresponding to this candidate can be evaluated. Hence, there are 20 classification rates associated with 20 candidates. The candidate providing the highest classification rate is chosen. This classification rate is recorded in the tables and so is its associated N_S in brackets; cf. the second row of Table 4.1, as well as Table 4.2 and Table 4.3. Apparently, the optimal N_S is classifier-dependent. A fixed global setting of N_S for all the four classifiers would be impractical. As a consequence, these three methods are very time-consuming. The factor $\varkappa$ required for the methods MIFS and MIFS-U, which has been mentioned in the introduction, can take values in $[0, 1]$ and its influence on the performance can be observed in Table 4.2 and Table 4.3. The results in both tables demonstrate another fact that even the choice of an optimal $\varkappa$ is classifier-dependent. Finally, it is obvious that these classification rates in Table 4.1, Table 4.2 and Table 4.3 corresponding to the methods mRMR, MIFS and MIFS-U are obtained in their best cases.

The feature selection methods RELFSS and mRMR do not depend on additional parameters. When classifier $\mathfrak{e} \in \mathfrak{E}$ is applied, their results in Table 4.1 are denoted as $\rho_{\text{RELFSS},\mathfrak{e}}$ and $\rho_{\text{mRMR},\mathfrak{e}}$. The feature selection methods MIFS and MIFS-U are subject to the setting of $\varkappa$. Thus, their results are denoted as $\rho_{\text{MIFS},\mathfrak{e}}(\varkappa)$, $\rho_{\text{MIFS-U},\mathfrak{e}}(\varkappa)$ with $\mathfrak{e} \in \mathfrak{E}$, and $\varkappa \in [0, 1]$, respectively. A classification performance gain indicator is defined to compare the MCRM-SFSA and MCRM-SFSG with the four reference methods mentioned above. Taking the classification rates of classifier $\mathfrak{e} \in \mathfrak{E}$ into consideration, the

classification performance gain indicator is given as follows:

$$\begin{aligned} q_{a,\mathfrak{e}}(\mathbf{\Gamma}_a) &= \operatorname{sgn}\left(\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a) - \mathfrak{p}_{\mathfrak{e}}\right), \text{ for MCRM-SFSA} \\ q_{g,\mathfrak{e}}(\mathbf{\Gamma}_g) &= \operatorname{sgn}\left(\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g) - \mathfrak{p}_{\mathfrak{e}}\right), \text{ for MCRM-SFSG} \\ \text{with} \quad & \mathfrak{p}_{\mathfrak{e}} = \max\left\{\rho_{\text{RELFSS},\mathfrak{e}}, \rho_{\text{mRMR},\mathfrak{e}}, \bar{\rho}_{\text{MIFS},\mathfrak{e}}, \bar{\rho}_{\text{MIFS-U},\mathfrak{e}}\right\}, \end{aligned} \tag{4.24}$$

where $\bar{\rho}_{\text{MIFS},\mathfrak{e}}$ and $\bar{\rho}_{\text{MIFS-U},\mathfrak{e}}$ denote the column-wise averages of $\rho_{\text{MIFS},\mathfrak{e}}(\varkappa)$, $\rho_{\text{MIFS-U},\mathfrak{e}}(\varkappa)$ in Table 4.2 and Table 4.3, respectively. In Fig. 4.11, the $q_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ with $\mathbf{\Gamma}_a \in \tilde{\mathbb{A}} = \{(\gamma_{a,W}, \gamma_{a,H})^T | 0 < \gamma_{a,W}, \gamma_{a,H}, 0 < \gamma_{a,W} + \gamma_{a,H} < 1\}$ of MCRM-SFSA are depicted. The $q_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$ corresponding to MCRM-SFSG with $\mathbf{\Gamma}_g \in \tilde{\mathbb{G}} = \{(\gamma_{g,W}, \gamma_{g,H})^T | 0 < \gamma_{g,W}, \gamma_{g,H}, 0 < \gamma_{g,W} + \gamma_{g,H} < 1\}$ are given in Fig. 4.12. The value of $\rho_{\text{Ref},\mathfrak{e}}$ for different classifiers can be found in the third row of Table 4.4.

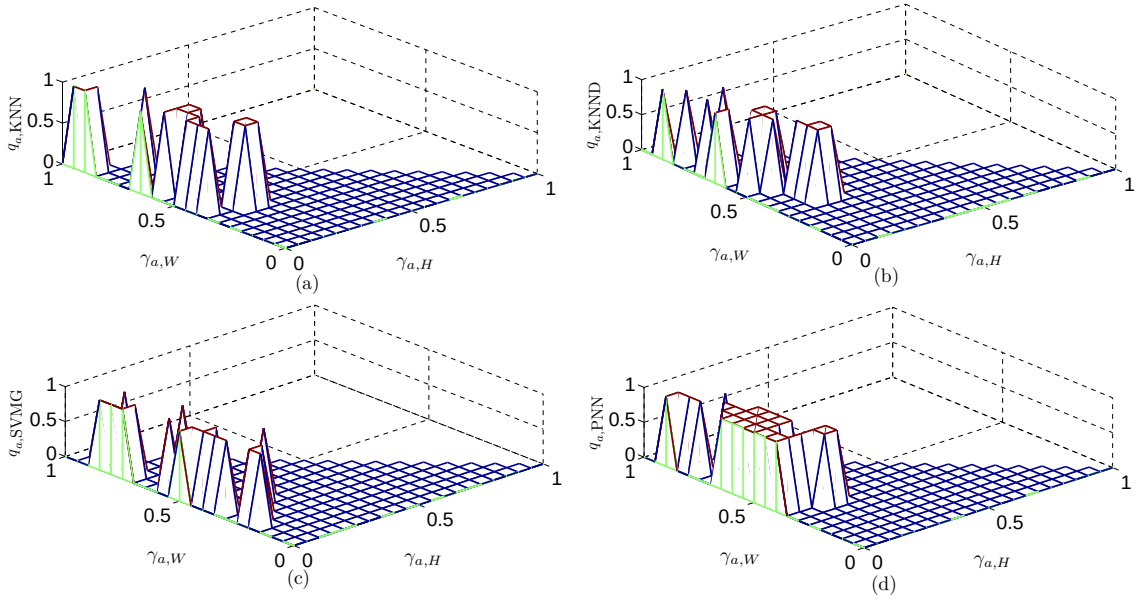


Figure 4.11. The $q_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ of various classifiers for the selections obtained by MCRM-SFSA: (a) $q_{a,\text{KNN}}(\mathbf{\Gamma}_a)$, (b) $q_{a,\text{KNND}}(\mathbf{\Gamma}_a)$, (c) $q_{a,\text{SVMG}}(\mathbf{\Gamma}_a)$ and (d) $q_{a,\text{PNN}}(\mathbf{\Gamma}_a)$.

The $q_{a,\mathfrak{e}}(\mathbf{\Gamma}_a) = 1$ indicates that the MCRM-SFSA can outperform the four reference methods when classifier $\mathfrak{e}$ is applied. Jointly observing the performances corresponding to MCRM-SFSA, $q_{a,\mathfrak{e}}(\mathbf{\Gamma}_a) = 1$ appears mainly in the region

$$\mathbb{A} = \left\{(\gamma_{a,W}, \gamma_{a,H})^T | 0.35 \leq \gamma_{a,W} \leq 0.85, 0 < \gamma_{a,H} < 0.2\right\}. \tag{4.25}$$

Similarly, $q_{g,\mathfrak{e}}(\mathbf{\Gamma}_g) = 1$ means that MCRM-SFSG outperforms the four reference methods. The case of $q_{g,\mathfrak{e}}(\mathbf{\Gamma}_g) = 1$ appears seldom when $\gamma_{g,H} > 0.6$. It occurs mainly in the region

$$\mathbb{G} = \left\{(\gamma_{g,W}, \gamma_{g,H})^T | 0.85 \leq \gamma_{g,W} + \gamma_{g,H} \leq 0.95, 0 < \gamma_{g,H} \leq 0.6, 0 < \gamma_{g,W} \leq 0.7\right\}. \tag{4.26}$$

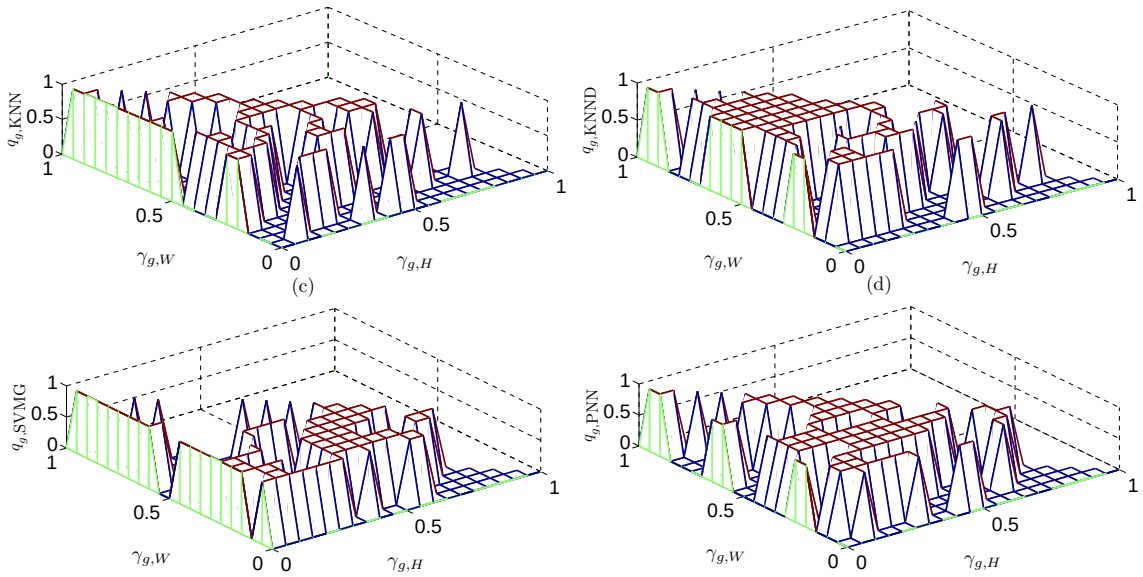


Figure 4.12. The $q_{a,\mathfrak{e}}(\mathbf{\Gamma}_g)$ of various classifiers for the selections obtained by MCRM-SFSG: (a) $q_{g,\mathrm{KNN}}(\mathbf{\Gamma}_g)$, (b) $q_{g,\mathrm{KNND}}(\mathbf{\Gamma}_g)$, (c) $q_{g,\mathrm{SVMG}}(\mathbf{\Gamma}_g)$ and (d) $q_{g,\mathrm{PNN}}(\mathbf{\Gamma}_g)$.

Both regions express the fact that neither the mRW nor the SE should be overemphasized when assessing the relevance of the feature selections. Comparing $q_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ in Fig. 4.11 with $q_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$ in Fig. 4.12, we find that the MCRM-SFSG can outperform the reference methods in more cases than the MCRM-SFSA.

Since the regions containing optimal $\mathbf{\Gamma}_a$ and $\mathbf{\Gamma}_g$ settings are found, the following discussion is constrained to the classification results that are obtained by using the features selected with $\mathbf{\Gamma}_a \in \mathbb{A}$ and $\mathbf{\Gamma}_g \in \mathbb{G}$. When classifier $\mathfrak{e}$ is employed, the average of $\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ over $\mathbb{A}$ is denoted as $\bar{\rho}_{a,\mathfrak{e}}$, and the average of $\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$ over $\mathbb{G}$ is denoted as $\bar{\rho}_{g,\mathfrak{e}}$. The $\bar{\rho}_{a,\mathfrak{e}}$ and $\bar{\rho}_{g,\mathfrak{e}}$ are shown in the first and second row of Table 4.4, respectively. Furthermore, the standard deviations of $\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ and $\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$ over $\mathbb{A}$ and $\mathbb{G}$ are given in the brackets of the first two rows as well, i.e. $\mathsf{s}_{a,\mathfrak{e}} = \sqrt{\frac{1}{|\mathbb{A}|}\sum_{\mathbf{\Gamma}_a \in \mathbb{A}} (\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a) - \bar{\rho}_{a,\mathfrak{e}})^2}$ and $\mathsf{s}_{g,\mathfrak{e}} = \sqrt{\frac{1}{|\mathbb{G}|}\sum_{\mathbf{\Gamma}_g \in \mathbb{G}} (\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g) - \bar{\rho}_{g,\mathfrak{e}})^2}$. The $\mathsf{s}_{a,\mathfrak{e}}$ and $\mathsf{s}_{g,\mathfrak{e}}$ are measures describing the performance dispersion of MCRM-SFSA and MCRM-SFSG, while the parameter settings, i.e. $\mathbf{\Gamma}_a$ and $\mathbf{\Gamma}_g$, change over $\mathbb{A}$ and $\mathbb{G}$, repsectively. Compared with the $\mathfrak{p}_{\mathfrak{e}}$ in the third row of Table 4.4, although the MCRM-SFSA and MCRM-SFSG do not steadily provide better results, they are more robust to parameter settings. Besides, the MCRM-SFSA and MCRM-SFSG are fast since there is no necessity to set N_S manually.

$\mathfrak{e}$	KNN	KNND	SVMG	PNN
$\bar{\rho}_{a,\mathfrak{e}}$	0.8688 (0.0157)	0.8788 (0.0176)	0.8834 (0.0136)	0.8788 (0.0189)
$\bar{\rho}_{g,\mathfrak{e}}$	0.8752 (0.0151)	0.8837 (0.0178)	0.8833 (0.0160)	0.8863 (0.0177)
$\mathfrak{p}_{\mathfrak{e}}$	0.8667	0.8762	0.8819	0.8676

Table 4.4. The $\bar{\rho}_{a,\mathfrak{e}}$ and $\bar{\rho}_{g,\mathfrak{e}}$ are given in the first and second row, respectively, and the $\mathsf{s}_{a,\mathfrak{e}}$ and $\mathsf{s}_{g,\mathfrak{e}}$ over $\mathbb{A}$ and $\mathbb{G}$ are given in the brackets. The best performance of reference methods, $\mathfrak{p}_{\mathfrak{e}}$, is recorded in the third row.

The feature selection's dependency on the classifiers should also be studied. The classification performances of features selected by MCRM-SFSA, MCRM-SFSG, RELFSS, mRMR, MIFS and MIFS-U can be summarized by the measures $\bar{\rho}_{a,\mathfrak{e}}$, $\bar{\rho}_{g,\mathfrak{e}}$, $\rho_{\text{RELFSS},\mathfrak{e}}$, $\rho_{\text{mRMR},\mathfrak{e}}$, $\bar{\rho}_{\text{MIFS},\mathfrak{e}}$ and $\bar{\rho}_{\text{MIFS-U},\mathfrak{e}}$, respectively. We calculate the range of variation δ_ρ of these measures over different classifiers, which describes the width of the variation interval of the classification rates over classifiers. A small value of δ_ρ indicates that different classifiers can provide similar classification rates using the same features selected by a certain method. In other words, the feature selection's dependency on classifiers is low. The δ_ρ of MCRM-SFSA and MCRM-SGSG are obtained by $\max_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{a,\mathfrak{e}} - \min_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{a,\mathfrak{e}}$ and $\max_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{g,\mathfrak{e}} - \min_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{g,\mathfrak{e}}$, respectively; cf. Table 4.4. Similarly, the δ_ρ of RELFSS, mRMR, MIFS and MIFS-U are obtained by $\max_{\mathfrak{e}\in\mathfrak{E}} \rho_{\text{RELFSS},\mathfrak{e}} - \min_{\mathfrak{e}\in\mathfrak{E}} \rho_{\text{RELFSS},\mathfrak{e}}$, $\max_{\mathfrak{e}\in\mathfrak{E}} \rho_{\text{mRMR},\mathfrak{e}} - \min_{\mathfrak{e}\in\mathfrak{E}} \rho_{\text{mRMR},\mathfrak{e}}$, $\max_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{\text{MIFS},\mathfrak{e}} - \min_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{\text{MIFS},\mathfrak{e}}$ and $\max_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{\text{MIFS-U},\mathfrak{e}} - \min_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{\text{MIFS-U},\mathfrak{e}}$, respectively.

The δ_ρ of different feature selection methods are presented in the first row of Table 4.5. Apparently, the δ_ρ of MCRM-SFSA and MCRM-SFSG are lower than those of the reference methods. It means that they can provide the feature selections that are suitable for a wider range of classifiers.

Method	RELFSS	mRMR	MIFS	MIFS-U	MCRM-SFSA	MCRM-SFSG
δ_ρ	0.0619	0.0191	0.0285	0.0324	**0.0146**	**0.0111**
$\mu_{\mathfrak{E}}$	0.8167	0.8714	0.8622	0.8648	**0.8775**	**0.8821**

Table 4.5. The comparison of the feature selection's dependency on classifiers. The dependency is expressed in terms of the range of the classification rate variation over the considered classifiers, δ_ρ. In the second row, $\mu_{\mathfrak{E}}$, the averaged classification rates over classifiers are also presented for different feature selection methods.

Moreover, the average classification rates over classifiers, $\mu_{\mathfrak{E}}$, are also given in the second row of Table 4.5. The $\mu_{\mathfrak{E}}$ of MCRM-SFSA, MCRM-SFSG, RELFSS, mRMR, MIFS and MIFS-U are calculated by $\frac{1}{|\mathfrak{E}|}\sum_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{a,\mathfrak{e}}$, $\frac{1}{|\mathfrak{E}|}\sum_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{g,\mathfrak{e}}$, $\frac{1}{|\mathfrak{E}|}\sum_{\mathfrak{e}\in\mathfrak{E}} \rho_{\text{RELFSS},\mathfrak{e}}$, $\frac{1}{|\mathfrak{E}|}\sum_{\mathfrak{e}\in\mathfrak{E}} \rho_{\text{mRMR},\mathfrak{e}}$, $\frac{1}{|\mathfrak{E}|}\sum_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{\text{MIFS},\mathfrak{e}}$ and $\frac{1}{|\mathfrak{E}|}\sum_{\mathfrak{e}\in\mathfrak{E}} \bar{\rho}_{\text{MIFS-U},\mathfrak{e}}$, respectively. The analysis of $\mu_{\mathfrak{E}}$ in Table 4.5 might

lead to the conclusion that the improvements provided by MCRM-SFSA and MCRM-SFSG are not significant. However, with such a reasoning one would ignore the fact that the performances for the reference methods, i.e. mRMR, MIFS and MIFS-U, are obtained in their best cases. Considering their classification rates in the second row of Table 4.1, Table 4.2 and Table 4.3, the feature sets used for classification are optimal with respect to the classification performance of each individual classifier, and these optimal feature sets associated with each individual classifier are usually different. In contrast, the feature set obtained with MCRM-SFSA or MCRM-SFSG for a certain parameter setting (either $\mathbf{\Gamma}_a$ or $\mathbf{\Gamma}_g$) is equivalently utilized by all classifiers in $\mathfrak{E}$. This feature set is not exclusively chosen based on the classification performance of a certain classifier, and it can be suboptimal for some individual classifiers. From this point of view, when MCRM-SFSA and MCRM-SFSG are applied, the performances presented in the second row of Table 4.5 are not obtained with their optimal configurations. Hence, the comparison of $\mu_{\mathfrak{E}}$ only demonstrates that the performances of MCRM-SFSA and MCRM-SFSG are, at least, not worse than the performances of the reference methods even for suboptimal settings.

4.5 Conclusions

In this chapter, we deal with the feature selection that takes place in the system design phase. The results of the feature selection would be saved and utilized to instruct the feature extraction in the object classification phase. A sophisticated filter method using a novel feature relevance measure is proposed to select the most relevant features out of the set that contains the features described in Chapter 3. This feature relevance measure, i.e. composite relevance measure, simultaneously takes the mutual information, the Shannon entropy and the modified Relief weight into consideration. Both linear and nonlinear combinations of these measures are considered. The mutual information is used to supervise the sufficiency of the selection. The consideration of Shannon entropy in the composite relevance measure is important to avoid both overfitting and underfitting. The modified Relief weight is proposed to help find an optimal feature selection among multiple sufficient feature sets. Since a complete search of all the possible combinations of features leads to an NP-hard problem, a heuristic method is adopted to construct the filter methods MCRM-SFSA and MCRM-SFSG.

The MCRM-SFS is applied to select the features for the classification of underwater targets. The regions for optimal parameter settings in which the MCRM-SFS can mostly outperform the reference methods are found. None of the mutual information, the modified Relief weight and the Shannon entropy can be overemphasized in the construction of the composite relevance measure. Moreover, it can be concluded that the nonlinear combination of Shannon entropy, mutual information and modified Relief weight can better evaluate the feature relevance. Compared with those methods in the literature the MCRM-SFS is much faster since there is no requirement of a manual setting of the number of selected features. In addition, the performance variations of the features selected by MCRM-SFS over different classifiers are the lowest. In other words, the MCRM-SFS is able to provide the features which are suitable to a wide range of classifiers. This advantage of the composite relevance measure can simplify the design of an automatic detection and automatic classification system to a great extent, since it allows to decouple the optimal feature selection and optimal classifier selection process in two consecutive steps.

Chapter 5

Object Classification Using Ensemble Learning

In this chapter, a reliable classification of MLO is elaborated, i.e. the prediction about the types of MLO. The features selected by the method introduced in Chapter 4 are employed. It had been observed in many numerical studies that individual classifiers, e.g. [60,134,136–139], could be improved to a certain degree. Keller *et al.* [137] have incorporated the fuzzy set theory into the k-nearest neighbor technique [136] to develop a fuzzy k-nearest neighbor algorithm. Vert *et al.* [134] have improved the support vector machine with their sophisticated kernels. As an improved version of the probabilistic neural network given in [60], Streit *et al.* have proposed a generalized Fisher training model in [138]. Instead of a Parzen probability density estimation [65], they used a Gaussian mixture model to approximate the probability density function. Thus, the number of nodes in the pattern layer can be reduced. In addition, Zhang has summarized some of the most important developments in neural network classification research in [139]. However, possibly none of them is perfect due to the complexity of underwater targets displayed in sonar images. Furthermore, the sets of patterns misclassified by different classifiers would not necessarily overlap. These observations motivated the recent interest in the topic of ensemble learning. The ensemble learning refers to those approaches that learn a target function by training a number of individual classifiers and fuse their outputs. The complementariness among the outputs of different classifiers, which can be modeled as information sources, is able to be utilized to promote the classification accuracy. The Dempster-Shafer theory has been demonstrated to be very useful to manage the uncertainty in the information obtained from diverse sources in Chapter 2, and it is also adopted in this chapter to fulfill the joint consideration of the classification results provided by different classifiers. This adoption initiates a new direction for the development of reliable ADAC systems devoted to target recognition in SAS imagery.

Various classifier combination schemes have been devised and it has been indicated that some of them consistently outperform a single best classifier, e.g. [66–68, 74]. There are two very popular structures for the design of ensemble learning schemes, namely the *multistage topology* and the *parallel topology*. The multistage topology [75–80] has gained great attention for a long time due to its efficiency, whereby objects are classified by simple classifiers using small sets of simple features in combination with reject options on individual stages. The parallel topology, e.g. [69–71, 73, 81–85], depicted

in Fig. 5.1 is also widely applied in ensemble learning because of its robustness. In this thesis, we adopt this topology. This approach applied to the fusion of classifiers depends on the outputs of classifiers. Generally speaking, the output information that

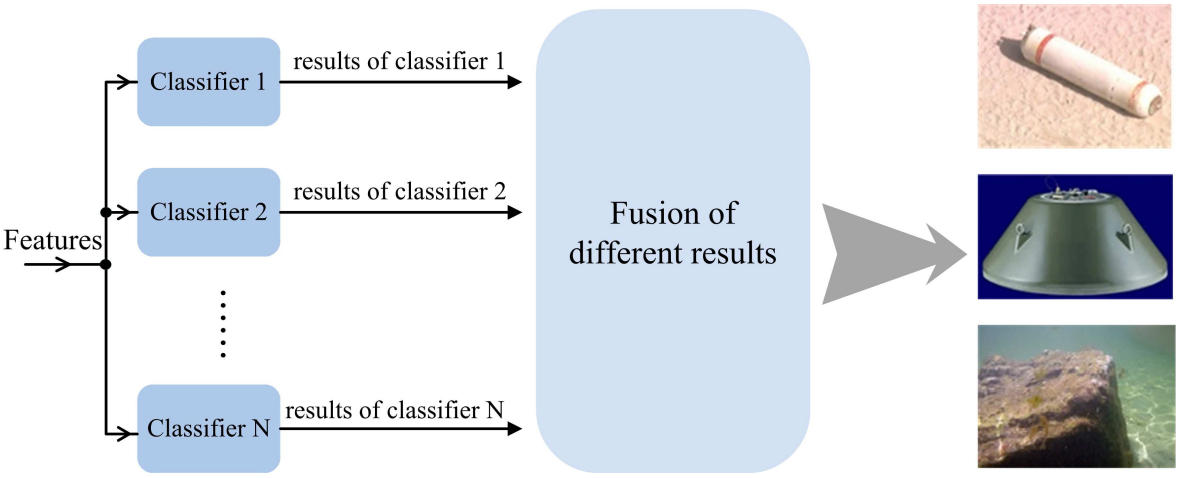


Figure 5.1. An ensemble learning scheme using the parallel topology.

various classifiers supply can be divided into three levels [70]:

1. **The abstract level**: A classifier only provides a unique index $c \in \boldsymbol{\mathcal{C}}$.
2. **The rank level**: A classifier arranges all the class indices belonging to $\boldsymbol{\mathcal{C}}$ in a queue with the index at the top being the first choice.
3. **The measurement level**: A classifier assigns each index in $\boldsymbol{\mathcal{C}}$ a confidence value to denote the degree of support to the hypothesis that an object has the class index.

The abstract level and the rank level do not provide the amount of support behind the hypothesis that a MLO could be assigned with a certain class index. When only outputs on the abstract level or rank level are available, a majority vote, e.g. [81–84], can be adopted to fuse these outputs. When the classifiers provide results on the measurement levels, an average or some other linear combination scheme of the confidence values has been proposed [69–71]. Recently, more sophisticated techniques, such as Dempster-Shafer theory (DST) techniques [70, 73, 85, 140, 141], have also been widely used. An important issue related to DST techniques is how to set the basic belief assignment (BBA). Zhang *et al.* [85] and Xu *et al.* [70] used empirical knowledge to assign belief portions, and Rogova [73] suggested the distance between the reference vector and the object vector to be the basis for BBA. However, the choice of a reference vector is

not an easy task since the cluster of one class is not necessarily unique in the feature space. In particular, Mignotte *et al.* [140] and Fawcett *et al.* [141] applied the DST to object classification in sonar imagery. Mignotte *et al.* derived their BBA from a confusion matrix. Fawcett *et al.* introduced three kinds of specifications for BBA. Two of them required empirical knowledge, which may limit the extent of their applications. Although the third BBA specification was nonempirical, the obtained BBA was very intricate, and it could make the combination of BBAs computationally expensive.

In this chapter, an ensemble learning scheme using DST is proposed for mine type classification. In the derivation of BBA, the following two parts are considered. One is the support to the hypotheses provided by classifiers and the other one is the measure quantifying the reliability of the classifiers themselves. The first part (object part) is usually unequal for different test objects and the second one (classifier part) is fixed for each classifier. Hence, the belief assigned to the hypotheses by the BBA is the product of the object part and the classifier part. To our best knowledge, it is the first time that DST is applied in SAS imagery.

This chapter is organized as follows. Sec. 5.1 describes the simple nontrainable combiners as well as the combiner using the DST technique proposed by Xu *et al.* The proposed model of multiclassifier combination in the framework of DST is presented in Sec. 5.2. The classification results can be found in Sec. 5.3.

5.1 Review of Classifier Combination Approaches Using Parallel Topology

5.1.1 Simple Nontrainable Combiners

As already defined in the previous chapter, $\boldsymbol{\mathcal{C}} = \{c_1, ...c_{N_c}\}$ contains all the class labels, and a classifier is denoted by $\mathfrak{e} \in \mathfrak{E}$, where $\mathfrak{E}$ is the set of all the implemented classifiers (base classifiers). For a test object, let $\mathfrak{y}_{\mathfrak{e}}(c_n)$ be the support provided by classifier $\mathfrak{e}$ to the hypothesis that this test object is assigned with label $c_n \in \boldsymbol{\mathcal{C}}$. The class label c_n of a test object is given by

$$c = \arg\max_{c_n \in \boldsymbol{\mathcal{C}}} \mathfrak{f}(c_n), \tag{5.1}$$

where $\mathfrak{f}$ is a combination function. The combination function $\mathfrak{f}$ can be chosen in many different forms. The most popular choices are:

- Average rule:

$$\mathfrak{f}(c_n) = \frac{1}{|\mathfrak{E}|} \sum_{\mathfrak{e} \in \mathfrak{E}} \mathfrak{y}_{\mathfrak{e}}(c_n). \tag{5.2}$$

- Maximum rule:

$$\mathfrak{f}(c_n) = \max_{\mathfrak{e} \in \mathfrak{E}} \mathfrak{y}_{\mathfrak{e}}(c_n). \tag{5.3}$$

- Median rule:

$$\mathfrak{f}(c_n) = \underset{\mathfrak{e} \in \mathfrak{E}}{\text{median}}\, \mathfrak{y}_{\mathfrak{e}}(c_n). \tag{5.4}$$

- Product rule:

$$\mathfrak{f}(c_n) = \prod_{\mathfrak{e} \in \mathfrak{E}} \mathfrak{y}_{\mathfrak{e}}(c_n). \tag{5.5}$$

The above-mentioned schemes are called nontrainable combiners, because other than the training of individual base classifiers there are no extra parameters that need to be trained. The ensemble is ready for operation as soon as the base classifiers are trained. Hence, due to their simplicity the nontrainable combiners have been widely used for a long time.

5.1.2 Combination of Classifiers Using the Method of Xu *et al.*

The DST adopted in this chapter has been introduced in Chapter 2.3.2. The combination method proposed by Xu *et al.* in [70] works on the abstract level of the classifier output. In their framework, given a test object, it can be classified to class $c \in \boldsymbol{\mathcal{C}} \bigcup \{\mathcal{C}_{\text{reject}}\}$, where the $\mathcal{C}_{\text{reject}}$ denotes that the classifier has no idea from which class the test object comes. Accordingly, there is a performance measure that quantifies the fraction of the object being classified to $\mathcal{C}_{\text{reject}}$, i.e. rejection rate. According to their definition, the substitution rate denotes the fraction of the falsely classified objects, and we have $\rho + \mathfrak{r}_s + \mathfrak{r}_r = 1$. Their approach works in the circumstance that the identification rate (ρ), substitution rate ($\mathfrak{r}_s$) and rejection rate ($\mathfrak{r}_r$) of individual classifiers are available.

When classifier $\mathfrak{e} \in \mathfrak{E}$ provides a prediction that an object belongs to class $c_n \in \boldsymbol{\mathcal{C}}$, the BBA is given as

$$\begin{cases} \mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\Omega}) = \rho, & \text{for} \quad \boldsymbol{\Omega} = \{c_n\}, \\ \mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\Omega}) = \mathfrak{r}_s, & \text{for} \quad \boldsymbol{\Omega} = \boldsymbol{\mathcal{C}} \backslash \{c_n\}, \\ \mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\Omega}) = \mathfrak{r}_r, & \text{for} \quad \boldsymbol{\Omega} = \{\mathcal{C}_{\text{reject}}\}. \end{cases} \tag{5.6}$$

When classifier $\mathfrak{e}$ classifies the object as class $\mathcal{C}_{\text{reject}}$, the BBA of Xu *et al.* has only one focal, $\mathfrak{b}_{\mathfrak{e}}(\{\mathcal{C}_{\text{reject}}\}) = 1$. This formalism of BBA considers only the classifier part which provides the overall performance information of individual classifiers. The information specifically correlated to individual objects is overseen.

5.2 A Novel Proposal for the Classifier Combination in DST

5.2.1 The Construction of Basic Belief Assignment

In order to derive our ensemble learning in the framework of DST, the classification result of an individual classifier is viewed as a piece of evidence. Subsequently, a BBA is induced from this piece of evidence. The BBA proposed in this paper is constructed of two parts, i.e. the object part and the classifier part. The object part, which is a non-empirical part, gives the information about how much support a classifier can provide to a certain hypothesis out of the set $\boldsymbol{\mathcal{C}}$. The classifier part, which depends on empirical knowledge, quantifies the quality of the judgment given by a classifier. Hence, the support provided by the object part should be discounted by the classifier part. Let $\mathfrak{E}$ be the set of all the implemented classifiers. Obviously, there are $M_{\mathfrak{E}} = |\mathfrak{E}|$ BBAs induced from the classification results of the implemented classifiers. The classifier part and object part associated with classifier $\mathfrak{e} \in \mathfrak{E}$ are denoted as $\mathfrak{c}_{\mathfrak{e}}$ and $\mathfrak{o}_{\mathfrak{e}}$, respectively. Therefore, the BBA induced from the result of classifier $\mathfrak{e}$ is given as follows:

$$\mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\Omega}) = \begin{cases} \mathfrak{c}_{\mathfrak{e}}\mathfrak{o}_{\mathfrak{e}}(\boldsymbol{\Omega}), & \text{for} \quad \boldsymbol{\Omega} = \{c_n\} \\ 1 - \sum\limits_{c_n \in \boldsymbol{\mathcal{C}}} \mathfrak{b}_{\mathfrak{e}}(\{c_n\}), & \text{for} \quad \boldsymbol{\Omega} = \boldsymbol{\mathcal{C}} \\ 0, & \text{otherwise} \end{cases} \tag{5.7}$$

where $\mathfrak{c}_{\mathfrak{e}}, \mathfrak{o}_{\mathfrak{e}}$ are the classifier part and object part corresponding to classifier $\mathfrak{e}$ respectively, $c_n \in \boldsymbol{\mathcal{C}}$, $0 \leq \mathfrak{c}_{\mathfrak{e}}\mathfrak{o}_{\mathfrak{e}}(\boldsymbol{\Omega}) \leq 1$ and $0 \leq \sum_{c_n \in \boldsymbol{\mathcal{C}}} \mathfrak{b}_{\mathfrak{e}}(\{c_n\}) \leq 1$. In our application, there are four kinds of classifiers adopted and they were already clarified in Equation (4.22). Now, based on the empirical results obtained for each classifier $\mathfrak{e} \in \mathfrak{E}$, the classifier part and object part are specified as follows.

The classifier part, $\mathfrak{c}_{\mathfrak{e}}$, requires the knowledge of the classifier's performance gained from the experimental observations. Intuitively without any *a priori* knowledge about the performance of individual classifiers, the $\mathfrak{c}_{\mathfrak{e}}$ of all classifiers is set equally, e.g. $\mathfrak{c}_{\mathfrak{e}} = 1, \forall \mathfrak{e} \in \mathfrak{E}$. When prior knowledge about the classifier's performance is available, it

enables a more reasonable setting of the classifier part. The details about the setting will be given in Sec. 5.3.

The object part, $\mathfrak{o}_{\mathfrak{e}}$, reveals the support assigned by classifier $\mathfrak{e}$ to the hypotheses out of $\boldsymbol{\mathcal{C}}$, where $\mathfrak{o}_{\mathfrak{e}}(\boldsymbol{\Omega})$ specifies the support dedicated to hypothesis $\boldsymbol{\Omega}$. The construction of $\mathfrak{o}_{\mathfrak{e}}$ is described as follows.

- **KNN** : The $\mathfrak{o}_{\text{KNN}}$ depends on the number of training objects of different classes in the neighborhood, e.g. the support associated with class $c_n \in \boldsymbol{\mathcal{C}}$ is $\frac{m_{\mathfrak{o}}(\{c_n\})}{m_{\text{KNN}}}$, where m_{KNN} denotes the number of all training objects in the neighborhood of a test object and $m_{\mathfrak{o}}(\{c_n\})$ is the number of training objects in this neighborhood belonging to class c_n.

- **KNND** : The $\mathfrak{o}_{\text{KNND}}$ depends on the belief value generated by KNND. The KNND models the neighboring training objects as evidence and combines their BBAs by Dempster's rule to make the classification of a test object. Accordingly, the support associated with hypothesis $\{c_n\}$ depends on $\mathfrak{b}_{\mathfrak{o}}(\{c_n\}), c_n \in \boldsymbol{\mathcal{C}}$, where $\mathfrak{b}_{\mathfrak{o}}$ is the combined BBA obtained by combining the BBAs of nearest neighbors.

- **SVMG** : The $\mathfrak{o}_{\text{SVMG}}$ depends on the distance of the test object to the discrimination surface in the feature space, $d_{\mathfrak{o}}(\{c_n\}), c_n \in \boldsymbol{\mathcal{C}}$. The one-against-all scheme [142] is adopted. The distance $d_{\mathfrak{o}}(\{c_n\})$ to the discrimination surface that divides the feature space into class c_n and non-class c_n describes the support provided to the hypothesis $\{c_n\}$. A large distance indicates a great amount of support for the hypothesis $\{c_n\}$.

- **PNN** : The $\mathfrak{o}_{\text{PNN}}$ depends on the posterior probability provided by PNN, e.g. the support associated with hypothesis $\{c_n\}$ is $p_{\mathfrak{o}}(\{c_n\}), c_n \in \boldsymbol{\mathcal{C}}$.

Hence, the object part can be summarized as follows:

$$\mathfrak{o}_{\mathfrak{e}}(\{c_n\}) = \begin{cases} \frac{m_{\mathfrak{o}}(\{c_n\})}{m_{\text{KNN}}}, & \text{for} \quad \mathfrak{e} = \text{KNN}, \\ \frac{\mathfrak{b}_{\mathfrak{o}}(\{c_n\})}{\sum_{c_{n'} \in \boldsymbol{\mathcal{C}}} \mathfrak{b}_{\mathfrak{o}}(\{c_{n'}\})}, & \text{for} \quad \mathfrak{e} = \text{KNND}, \\ \frac{\exp(d_{\mathfrak{o}}(\{c_n\}))}{\sum_{c_{n'} \in \boldsymbol{\mathcal{C}}} \exp(d_{\mathfrak{o}}(\{c_{n'}\}))} & \text{for} \quad \mathfrak{e} = \text{SVMG}, \\ p_{\mathfrak{o}}(\{c_n\}), & \text{for} \quad \mathfrak{e} = \text{PNN}, \end{cases} \tag{5.8}$$

where m_{KNN} denotes the number of the training objects in the neighborhood of a test object for KNN, and $m_{\mathfrak{o}}(\{c_n\})$ is the number of training samples in the neighborhood belonging to class $c_n \in \boldsymbol{\mathcal{C}}$. As for the case of SVMG, there is a possibility that the

distance measure for a certain hypothesis is negative, cf. Fig. 5.2. When the discrimination surface divides the training data into class c_n and non-class c_n, the negative distance means that the test object and training objects belonging to the class c_n do not locate on the same side of the discrimination surface. However, the support to a hypothesis cannot be negative. Hence, it is transformed by an exponential function, cf. Equation (5.8). When the distance $d_{\mathfrak{o}}(\{c_n\})$ approaches negative infinity, the support to the hypothesis that this test object can be classified to class c_n becomes zero, as illustrated in Fig. 5.3.

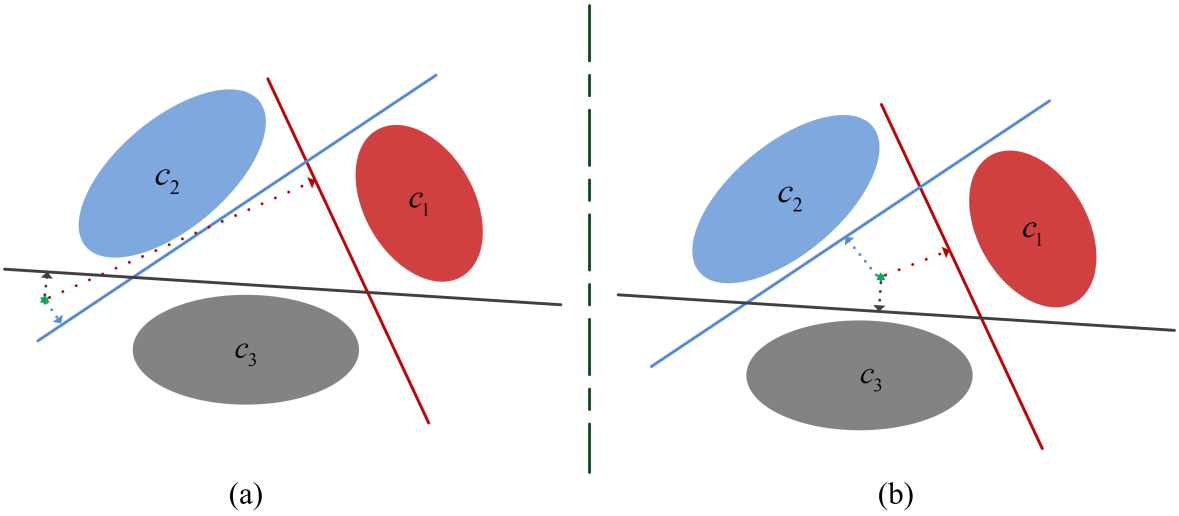


Figure 5.2. An illustration of object part for SVM. The training data is divided into three classes, $c_1, c_2, c_3 \in \boldsymbol{\mathcal{C}}$. The green star represents a test object. Each line denotes a discrimination surface and divides the training data into class c_n and non-class c_n for $c_n \in \boldsymbol{\mathcal{C}}$. In (a), although the test object has the longest distance to the red line, this distance is negative. The associated hypothesis has the least support. In (b), all the three distances are negative. In this case, the hypothesis corresponding to the least absolute distance has the greatest support.

It can be easily proven that $\sum_{c_n \in \boldsymbol{\mathcal{C}}} \mathfrak{o}_{\mathfrak{e}}(\{c_n\}) = 1$. If the prediction of classifier $\mathfrak{e}$ is 100% credible, the $\mathfrak{c}_{\mathfrak{e}}$ is set to 1. In this case, we can find that $\mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\mathcal{C}}) = 0$. Otherwise, we have $\mathfrak{c}_{\mathfrak{e}} < 1$. Accordingly, $\mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\mathcal{C}}) > 0$, where $\mathfrak{b}_{\mathfrak{e}}(\boldsymbol{\mathcal{C}}) > 0$ describes the degree to which one can not discriminate the hypothesis out of the set $\boldsymbol{\mathcal{C}}$.

5.2.2 The Application of Dempster's Rule and the Decision Rule

In Section 2.3.2, Dempster's rule is adopted to combine the BBAs obtained from the neighboring pixels. An effective combination scheme which was derived by Denoeux *et*

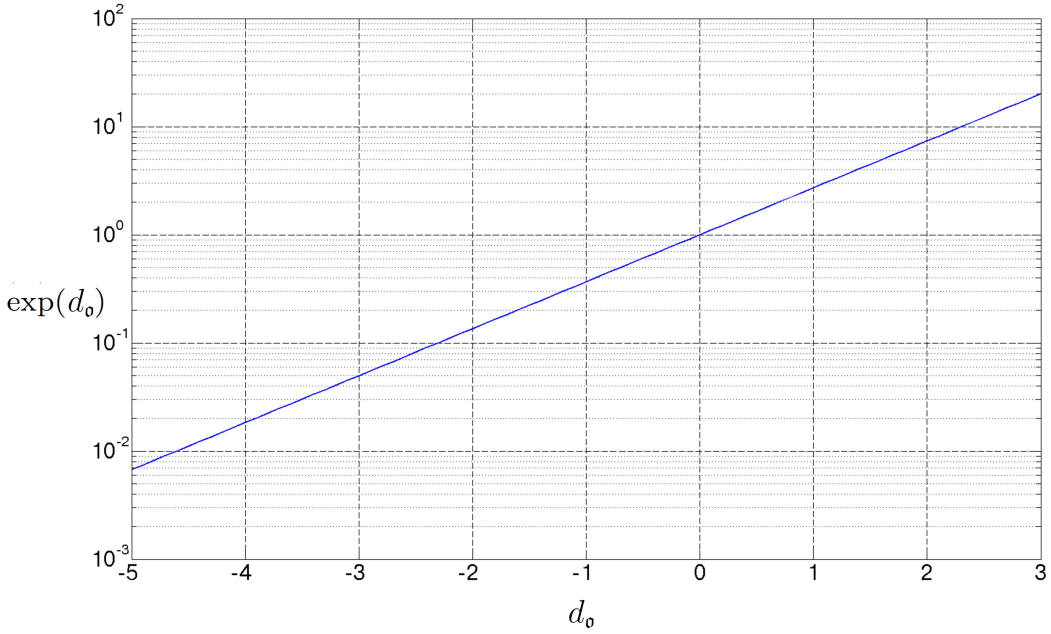


Figure 5.3. The exponential of distance $d_{\mathfrak{o}}$. It makes sure that the support is non-negative. When the $d_{\mathfrak{o}}$ approaches negative infinity, the support becomes zero.

al. [63] is implemented in order to alleviate the computation load involved in combining BBAs. However, it is designated to the simple BBA in Equation (2.60). The BBA (cf. Equation (5.7)) applied to the ensemble learning in this chapter is not a simple BBA. However, it is still the case that most of the elements in the power set $2^{\mathcal{C}}$ are not focals of this BBA. Hence, Dempster's rule can be simplified as follows. Let $\mathfrak{E}_n^m \subseteq \mathfrak{E}$ be the n-th subset that satisfies $|\mathfrak{E}_n^m| = m$ with $1 \leq n \leq \mathfrak{M}$ and $\mathfrak{M} = \binom{M_{\mathfrak{E}}}{m}$, and accordingly let $\bar{\mathfrak{E}}_n^m = \mathfrak{E} \backslash \mathfrak{E}_n^m$ denote its complementary set. The combined BBA, $\mathfrak{b}_{\oplus}$, is given as

$$\mathfrak{b}_{\oplus}(\mathbf{\Omega}) = \begin{cases} \frac{\mathfrak{b}'_{\oplus}(\mathbf{\Omega})}{\mathfrak{b}'_{\oplus}(\mathcal{C}) + \sum\limits_{c_i \in \mathcal{C}} \mathfrak{b}'_{\oplus}(\{c_i\})}, & \text{for } \mathbf{\Omega} = \{c_n\} \\ \frac{\mathfrak{b}'_{\oplus}(\mathcal{C})}{\mathfrak{b}'_{\oplus}(\mathcal{C}) + \sum\limits_{c_i \in \mathcal{C}} \mathfrak{b}'_{\oplus}(\{c_i\})}, & \text{for } \mathbf{\Omega} = \mathcal{C} \\ 0, & \text{otherwise} \end{cases} \tag{5.9}$$

where $\mathfrak{b}'_{\oplus}$ is defined as

$$\mathfrak{b}'_{\oplus}(\mathbf{\Theta}) = \begin{cases} \sum\limits_{m=1}^{M_{\mathfrak{E}}} \sum\limits_{n=1}^{\mathfrak{M}} \left(\prod\limits_{\mathfrak{e} \in \mathfrak{E}_n^m} \mathfrak{b}_{\mathfrak{e}}(\mathbf{\Theta}) \prod\limits_{\mathfrak{e}' \in \bar{\mathfrak{E}}_n^m} \mathfrak{b}_{\mathfrak{e}'}(\mathcal{C}) \right), & \text{for } \mathbf{\Theta} = \{c_n\} \\ \prod\limits_{\mathfrak{e} \in \mathfrak{E}} \mathfrak{b}_{\mathfrak{e}}(\mathcal{C}), & \text{for } \mathbf{\Theta} = \mathcal{C} \\ 0, & \text{otherwise} \end{cases} \tag{5.10}$$

The final decision on object classification can be obtained by maximizing the pignistic

probability, i.e.

$$c = \arg\max_{c_n \in \boldsymbol{\mathcal{C}}} \sum_{\boldsymbol{\Theta} \subseteq \boldsymbol{\mathcal{C}}} \mathfrak{b}_{\oplus}(\boldsymbol{\Theta}) \frac{|\{c_n\} \bigcap \boldsymbol{\Theta}|}{|\boldsymbol{\Theta}|}. \tag{5.11}$$

Since the focal elements of $\mathfrak{b}_{\oplus}$ are also either singletons $\{c_n\} \subset \boldsymbol{\mathcal{C}}$ or $\boldsymbol{\mathcal{C}}$ itself, Equation (5.11) can be simplified to

$$c = \arg\max_{c_n \in \boldsymbol{\mathcal{C}}} \mathfrak{b}_{\oplus}(\{c_n\}). \tag{5.12}$$

The complete classifier combination process can be summarized as follows, cf. Fig. 5.4.

1. Take the features provided by the step of feature extraction, which is guided by the results obtained with MCRM-SFS.
2. Run the classification by using the classifiers in $\mathfrak{E}$ and save their outputs.
3. Construct four BBAs ($\mathfrak{b}_{\epsilon}$) using the outputs of classifiers in $\mathfrak{E}$ according to Equation (5.7) and Equation (5.8).
4. Fuse the BBAs using Equation (5.9) and Equation (5.10) to obtain the $\mathfrak{b}_{\text{ensemble}}$.
5. Classify the test object according to the rule given in Equation (5.12).

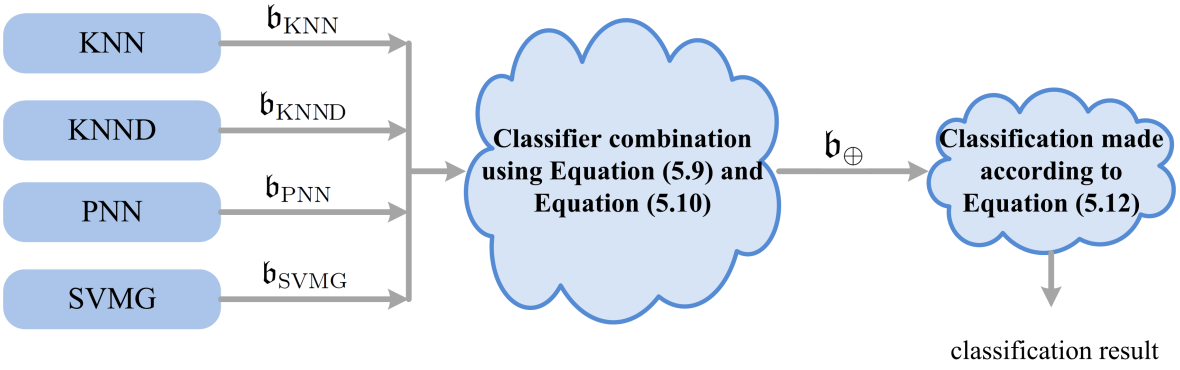


Figure 5.4. The illustration of the proposed ensemble learning scheme.

5.3 The Numerical Studies of Ensemble Learning

In this section, the database described in Chapter 4.4.3 is utilized to study the ability of ensemble learning in improving the classification results. We use the features provided

by the methods MCRM-SFSA and MCRM-SFSG. Rather than using an individual classifier in $\mathfrak{E}$, the classification in this subsection is made by ensemble learning. Given a parameter setting, either $\mathbf{\Gamma}_a$ or $\mathbf{\Gamma}_g$, features are selected and they are used for all the four classifiers in $\mathfrak{E}$. We obtain the final classification of test objects based on the combination of the results of these four classifiers. The combination scheme is either the one proposed in Sec. 5.2, or some other classifier combination scheme in the literature for the sake of comparison, cf. Sec. 5.1. The leave-one-out scheme is also adopted and repeated M times, so that all the M objects in the database have the associated classification results obtained by ensemble learning. Therefore, the performance of the ensemble learning schemes can also be evaluated by considering the classification rate in (4.23). The classification rate is denoted as $\rho_{a,\mathrm{en}}(\mathbf{\Gamma}_a)$ when MCRM-SFSA is employed to select features for the ensemble learning, and accordingly as $\rho_{g,\mathrm{en}}(\mathbf{\Gamma}_g)$ when MCRM-SFSG is used. The ensemble learning performance gain indicator is defined as follows

$$\begin{aligned} Q_{a,\mathrm{en}}(\mathbf{\Gamma}_a) &= \mathrm{sgn}\left(\rho_{a,\mathrm{en}}(\mathbf{\Gamma}_a) - \max_{\mathfrak{e}\in\mathfrak{E}} \rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)\right), \text{ for MCRM-SFSA,} \\ Q_{g,\mathrm{en}}(\mathbf{\Gamma}_g) &= \mathrm{sgn}\left(\rho_{g,\mathrm{en}}(\mathbf{\Gamma}_g) - \max_{\mathfrak{e}\in\mathfrak{E}} \rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)\right), \text{ for MCRM-SFSG.} \end{aligned} \tag{5.13}$$

The choice of $\mathfrak{c}_\mathfrak{e}$ requires *a priori* knowledge of the classifier's performance. In Sec. 4.4 the performances of individual classifiers are presented. We choose the average of the $\rho_{a,\mathfrak{e}}(\mathbf{\Gamma}_a)$ over $\tilde{\mathbb{A}}$ as the classifier part, when MCRM-SFSA is applied for feature selection,

$$\mathfrak{c}_{a,\mathfrak{e}} = \begin{cases} 0.8183, & \text{for } \mathfrak{e} = \mathrm{KNN}, \\ 0.8243, & \text{for } \mathfrak{e} = \mathrm{KNND}, \\ 0.8462, & \text{for } \mathfrak{e} = \mathrm{SVMG}, \\ 0.8134, & \text{for } \mathfrak{e} = \mathrm{PNN}. \end{cases} \tag{5.14}$$

When MCRM-SFSG is utilized for feature selection, the average of the $\rho_{g,\mathfrak{e}}(\mathbf{\Gamma}_g)$ over $\tilde{\mathbb{G}}$ is used,

$$\mathfrak{c}_{g,\mathfrak{e}} = \begin{cases} 0.8568, & \text{for } \mathfrak{e} = \mathrm{KNN}, \\ 0.8660, & \text{for } \mathfrak{e} = \mathrm{KNND}, \\ 0.8752, & \text{for } \mathfrak{e} = \mathrm{SVMG}, \\ 0.8609, & \text{for } \mathfrak{e} = \mathrm{PNN}. \end{cases} \tag{5.15}$$

We denote the case that *a priori* knowledge of the classifiers is known as $T1$. Moreover, we also choose $\mathfrak{c}_\mathfrak{e}$ equal for all the four classifiers to test the stability of our method when no *a priori* knowledge is available, and this case is denoted as $T2$. The quantities $Q_{a,\mathrm{en}}$ and $Q_{g,\mathrm{en}}$ of the proposed ensemble learning scheme are shown in Fig. 5.5 and Fig. 5.6, respectively. The ensemble learning performance gain indicator equal to 1 indicates the fact that the ensemble learning can improve the classification rates regarding individual classifiers. The results shown in Fig. 5.5 and Fig. 5.6 demonstrate that except for several settings, in both cases, i.e. for $T1$ and $T2$, the proposed scheme improves the

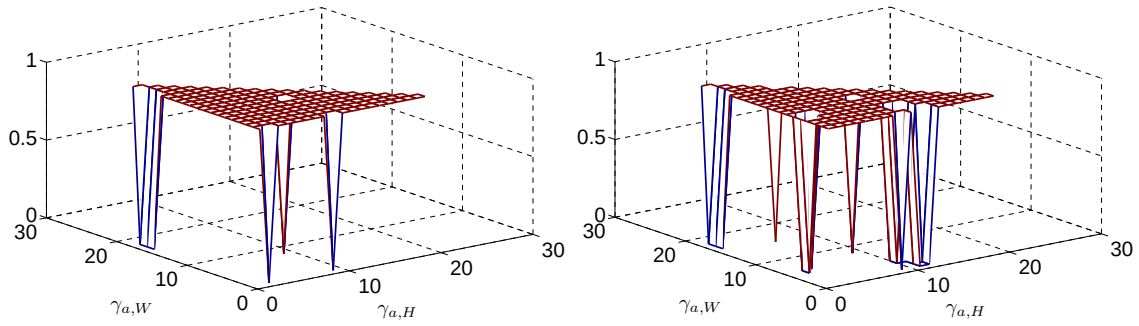


Figure 5.5. The $Q_{a,\mathrm{en}}(\mathbf{\Gamma}_a)$ obtained by the proposed ensemble learning scheme. Features are selected by MCRM-SFSA. The region without grids corresponds to the settings $\gamma_{a,W}+\gamma_{a,H}\geq 1$. (a) The $\mathfrak{c}_{a,\mathfrak{e}}$ chooses the setting $T1$ in (5.14), (b) the $\mathfrak{c}_{a,\mathfrak{e}}$ chooses the setting $T2$ with $\mathfrak{c}_{a,\mathfrak{e}}=1\ \forall \mathfrak{e}\in\mathfrak{E}$.

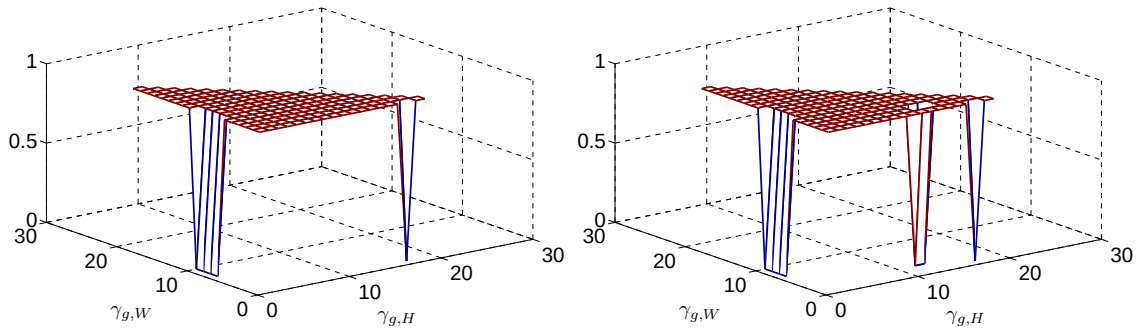


Figure 5.6. The $Q_{g,\mathrm{en}}(\mathbf{\Gamma}_g)$ obtained by the proposed ensemble learning scheme. Features are selected by MCRM-SFSG. The region without grids corresponds to the settings $\gamma_{g,W}+\gamma_{g,H}\geq 1$. (a) The $\mathfrak{c}_{g,\mathfrak{e}}$ chooses the setting $T1$ in (5.15), (b) the $\mathfrak{c}_{g,\mathfrak{e}}$ chooses the setting $T2$ with $\mathfrak{c}_{g,\mathfrak{e}}=1\ \forall \mathfrak{e}\in\mathfrak{E}$.

classification results provided by the classifiers in $\mathfrak{E}$. In other words, the ensemble learning scheme proposed in Sec. 5.2 is generally able to improve the classification performance of individual classifiers. Obviously, the proposed scheme works better in case $T1$. In reality, the classifier part is probably unknown *a priori*, and it has to be estimated. Therefore, the resulting classification performance could fall between those of $T1$ and $T2$.

Furthermore, quantitative analysis of the ensemble learning is presented. Since the optimal settings for $\mathbf{\Gamma}_a$ and $\mathbf{\Gamma}_g$ are available, in the following discussion ensemble learning schemes use the features selected corresponding to the parameter vectors $\mathbf{\Gamma}_a\in\mathbb{A}$ and $\mathbf{\Gamma}_g\in\mathbb{G}$. The averages of the $\rho_{a,\mathrm{en}}(\mathbf{\Gamma}_a)$ and $\rho_{g,\mathrm{en}}(\mathbf{\Gamma}_g)$ over $\mathbb{A}$ and $\mathbb{G}$ are considered. When the MCRM-SFSA is used for feature selection, the average classification rate over $\mathbb{A}$ is denoted as $\bar{\rho}_{a,\mathrm{en}}$. Similarly, when the MCRM-SFSG is used for feature selection, the average classification rate over $\mathbb{G}$ is written as $\bar{\rho}_{g,\mathrm{en}}$. The $\bar{\rho}_{a,\mathrm{en}}$ and $\bar{\rho}_{g,\mathrm{en}}$

obtained by different ensemble learning schemes are recorded in the second up to seventh row of Table 5.1, and the standard deviations of $\rho_{a,\mathrm{en}}(\mathbf{\Gamma}_a)$ and $\rho_{g,\mathrm{en}}(\mathbf{\Gamma}_g)$ over $\mathbb{A}$ and $\mathbb{G}$ are given in the brackets as well, i.e. $\mathsf{s}_{a,\mathrm{en}} = \sqrt{\frac{1}{|\mathbb{A}|}\sum_{\mathbf{\Gamma}_a\in\mathbb{A}}(\rho_{a,\mathrm{en}}(\mathbf{\Gamma}_a)-\bar{\rho}_{a,\mathrm{en}})^2}$ and $\mathsf{s}_{g,\mathrm{en}} = \sqrt{\frac{1}{|\mathbb{G}|}\sum_{\mathbf{\Gamma}_g\in\mathbb{G}}(\rho_{g,\mathrm{en}}(\mathbf{\Gamma}_g)-\bar{\rho}_{g,\mathrm{en}})^2}$.

Observing Table 5.1, the second row records the results of the proposed ensemble learning scheme which is operated in case $T1$. In rows three to seven of Table 5.1, the performance corresponding to ensemble learning using rules, such as average, median, maximum, product criterion [143] and the DST combination of Xu *et al.* [70], are depicted. The comparison between the proposed ensemble learning scheme and those schemes in the literature shows that the proposed ensemble learning scheme operated in case $T1$ has the best performance. Considering the $\mathsf{s}_{a,\mathrm{en}}$ and $\mathsf{s}_{g,\mathrm{en}}$ in the brackets, the performance dispersions of the proposed ensemble learning scheme are also marginal, while $\mathbf{\Gamma}_a$ and $\mathbf{\Gamma}_g$ change over $\mathbb{A}$ and $\mathbb{G}$, respectively. The results in the first row of Table 5.1 represent the best average classification rates which can be offered by an individual classifier out of $\mathfrak{E}$, i.e. $\max_{\mathfrak{e}\in\mathfrak{E}}\bar{\rho}_{a,\mathfrak{e}}$ and $\max_{\mathfrak{e}\in\mathfrak{E}}\bar{\rho}_{g,\mathfrak{e}}$. Comparing the results in the first row with those in the second row, we find that the proposed ensemble learning scheme can provide a significant performance gain.

Method Description	$\mathbb{A}$	$\mathbb{G}$
The best classification rate over various classifiers	0.8834	0.8863
Proposed ensemble learning scheme with $T1$	**0.9063** (0.0055)	**0.9147** (0.0049)
Ensemble learning using the average rule	0.8704 (0.0165)	0.8915 (0.0154)
Ensemble learning using the maximum rule of classifier combination	0.7955 (0.0110)	0.7433 (0.0104)
Ensemble learning using the median rule of classifier combination	0.8789 (0.0143)	0.8944 (0.0132)
Product rule of classifier combination	0.8796 (0.0210)	0.8925 (0.0185)
Ensemble learning scheme of Xu *et al.*	0.8825 (0.0089)	0.8976 (0.0077)

Table 5.1. The comparison of classification rates. The first row presents the best average classification rates offered by a single classifier out of $\mathfrak{E}$, i.e. $\max_{\mathfrak{e}\in\mathfrak{E}}\bar{\rho}_{a,\mathfrak{e}}$ and $\max_{\mathfrak{e}\in\mathfrak{E}}\bar{\rho}_{g,\mathfrak{e}}$. The quantities $\bar{\rho}_{a,\mathrm{en}}$ and $\bar{\rho}_{g,\mathrm{en}}$ of different classifier combination schemes are recorded in the second up to the seventh row, and the $\mathsf{s}_{a,\mathrm{en}}$ and $\mathsf{s}_{g,\mathrm{en}}$ over $\mathbb{A}$ and $\mathbb{G}$ are also presented in the brackets.

Besides, without *a priori* knowledge, the classifier part is set equally for different

MCRM-SFSA	$\mathfrak{c}_{a,\mathfrak{e}} = 1$	$\mathfrak{c}_{a,\mathfrak{e}} = 0.8$	$\mathfrak{c}_{a,\mathfrak{e}} = 0.6$	$\mathfrak{c}_{a,\mathfrak{e}} = 0.4$	$\mathfrak{c}_{a,\mathfrak{e}} = 0.2$
$\bar{\rho}_{a,\text{en}}$	0.8861 (0.0039)	0.8883 (0.0038)	0.8886 (0.0035)	0.8882 (0.0036)	0.8879 (0.0034)
MCRM-SFSG	$\mathfrak{c}_{g,\mathfrak{e}} = 1$	$\mathfrak{c}_{g,\mathfrak{e}} = 0.8$	$\mathfrak{c}_{g,\mathfrak{e}} = 0.6$	$\mathfrak{c}_{g,\mathfrak{e}} = 0.4$	$\mathfrak{c}_{g,\mathfrak{e}} = 0.2$
$\bar{\rho}_{g,\text{en}}$	0.9095 (0.0028)	0.9093 (0.0027)	0.9093 (0.0028)	0.9095 (0.0026)	0.9101 (0.0030)

Table 5.2. The quantities $\bar{\rho}_{a,\text{en}}$, $\bar{\rho}_{g,\text{en}}$ of the proposed ensemble learning scheme with classifier parts set as $T2$ are presented, and their $\mathsf{s}_{a,\text{en}}$, $\mathsf{s}_{g,\text{en}}$ are given in the brackets as well.

classifiers. The $\bar{\rho}_{a,\text{en}}$, $\bar{\rho}_{g,\text{en}}$ of the proposed ensemble learning scheme operated in case $T2$ are recorded in Table 5.2, and the associated $\mathsf{s}_{a,\text{en}}$, $\mathsf{s}_{g,\text{en}}$ are also given in the brackets. In the case of $T2$, it is observed that changing the values of the classifier parts has little influence on the performance of the proposed ensemble learning scheme. Although the performance is poorer than that obtained with the settings of case $T1$, it is still better than the standard combination methods recorded in rows three to six of Table 5.1.

5.4 Conclusions

In this chapter, a reliable ensemble learning scheme in the framework of Dempster-Shafer theory is developed to fulfill the task of object classification. This approach utilizes the outputs of individual classifiers as the information sources. A reasonable belief structure considering both classifier part and object part has been proposed. The classifier part containing the empirical knowledge about a classifier's performance can correct the support provided by this classifier to a certain hypothesis, i.e. the object part. Dempster's rule has been chosen to combine the BBAs induced by various classifiers. However, this kind of pairwise combination is time-consuming. In order to accelerate the combination process, a modified combination rule is derived. It is faster and can combine all the BBAs at once.

The proposed ensemble learning scheme is applied to the last step of the ADAC system, i.e. the classification of underwater objects. The results of the numerical studies demonstrate two facts about this approach. Firstly, the proposed ensemble learning scheme draws a performance gain compared with the results of individual classifiers. Secondly, it also provides better classification rates than those reference schemes using parallel topology in the literature. Moreover, the comparison between the settings of $T1$ and $T2$ shows that the incorporation of correct *a priori* knowledge about the classifier's performance is advantageous. However, it is also proven that the proposed ensemble learning scheme with a blind setting of classifier part is able to stably offer satisfying classification results. This characteristic widens the range of its application.

Chapter 6

Conclusions and Future Work

In this thesis, the problem of underwater mine classification in synthetic aperture sonar imagery has been considered. The automatic detection and automatic classification system is adopted to solve this problem. A modified expectation-maximization approach is applied to the image segmentation in regions of interest and the spatial correlation between pixels is tackled with Dempster-Shafer theory based clustering. In object classification, two issues have been mentioned, i.e. a choice of optimal features out of the complete feature set and a suitable ensemble learning scheme that combines the outputs of individual classifiers using parallel topology. The focuses have been set on making advances in the step of feature selection and improving the performance of the ensemble learning scheme.

The summary and main conclusions of the methods proposed in this thesis are provided in Section 6.1. The Section 6.2 presents an outlook for possible future works associated with the proposed methods.

6.1 Conclusions

6.1.1 The Dempster-Shafer Theory Supported EM Approach for Sonar Imagery Segmentation

In the area of image segmentation, an expectation-maximization approach assisted with Dempster-Shafer theory based clustering is developed. It provides reliable image segmentation results so that an extraction of geometrical features with fewer errors in synthetic aperture sonar imagery becomes possible. We extend the generalized expectation-maximization approach of Delignon *et al.* by substituting its mixture model with the one proposed by Sanjay-Gopal *et al.* In addition, the Peason system is also incorporated, and the mixture model is no longer constrainted to, for instance, the Gaussian mixture or the gamma mixture. The selection of optimal distribution types for individual classes can be automatically determined. The resulting model is more flexible in approximating the statistics of the sonar imagery. Furthermore, a Dempster-Shafer theory based clustering technique is incorporated to remove the clutters. We have proposed a belief structure to catch the information provided by the evidence in

the neighborhood. This belief structure considers not only the amount of the belief that the evidence can provide, but also the quality of the evidence. The implausible information existing in the neighborhood is not considered.

The proposed algorithm remarkably reduces the clutters in the background region of the sonar images, while preserving the shape of the objects. In addition, an improvement in the efficiency of this expectation-maximization approach becomes notable with the increasing of image size.

6.1.2 The Filter Method for Feature Selection Using a Novel Relevance Measure

In feature selection, the problem of selecting optimal features is considered. A novel feature relevance measure is proposed, which is a combination of the Shannon entropy, the mutual information and the modified Relief weight. In order to suppress the influence of outliers, the modified Relief weight adopts a distance measure with active rejection. In contrast to the original Relief weight, this modified Relief weight is not only applicable to individual features but also feature sets. Both arithmetic average and geometric average of these three measures are studied. Furthermore, another measure called sufficiency is developed to supervise the sufficiency of the feature selection and serves as a stopping criterion of the selection process.

The results of the numerical studies indicate three points. First of all, the proposed filter method can significantly accelerate the selection process since the searching of optimal cardinality of the feature selection is no longer required. Secondly, the selected features have a wider generalizability over different classifiers. Finally, the performance of the selected features is superior to that of the features obtained by the methods in the literature.

6.1.3 Classifier Combination in the Framework of Dempster-Shafer Theory

In combination, we introduced a Dempster-Shafer theory based ensemble learning scheme. It works on the measurement level to combine the information provided by individual classifiers. Compared with the methods using Dempster-Shafer theory based techniques in the literature, it includes not only the classifier part but also the object part in the design of basic belief assignment.

The proposed classifier combination scheme allows a performance gain over the classification results of individual classifiers, and it significantly enhances the reliability of the ensemble learning even when the prior knowledge about the classifiers' performance is unknown, i.e. the classifier parts for different classifiers are set equal.

6.2 Future Work

6.2.1 Image Segmentation

- **Prefiltering.** A lot of work has been done in the area of prefiltering images to reduce noise. In the last twenty years, methods like diffusion based smoothing filter [144], wavelets filter [145], bilateral filter [146], non-local means filter [147] and block-matching and 3D filter [148] have been proposed to improve the image quality. More specifically, the approaches in [149–152] have been applied to sonar imagery. The main challenge in underwater mine detection and classification is raised by the high amount of noise in the sonar imagery. Thus, a joint prefiltering and segmentation scheme could help to improve the final results.

- **Initialization of EM approach.** The problem of optimal initialization associated with the unsupervised segmentation is still open. The scheme proposed by Fandos *et al* in [113] has provided a satisfactory result in our application. However, the generalization to other applications should be studied and a generally optimal initialization scheme would be required.

6.2.2 Feature Selection

- **Sophisticated search scheme.** In the MCRM-SFSA and MCRM-SFSG, the heuristic scheme of sequential forward search is adopted because of its efficiency. In the last thirty years, many modifications have been discussed in the literature, e.g. sequential backward search, Plus-L minus-R search [153], bidirectional search and floating search [154]. They have achieved successes in some specific applications. However, in general cases none of them can guarantee optimal solutions. The combination of a composite relevance measure with alternative search scheme may help to improve the selection results.

- **Optimization of parameter settings.** The generalizability of the optimal parameter settings of the composite relevance measure to other applications should be further studied. Some optimization criteria, e.g. [155], could be applied to the search for optimal parameter settings for various applications.

6.2.3 Classifier Combination

- **Online processing.** The proposed combination scheme is only semi-online, and it does not incorporate any information about the real-time performance of individual classifiers. The knowledge associated with the classifier part is obtained by previous applications, and this knowledge could be improper for the current application. Taking the current performance of individual classifiers as feedback may be helpful to rectify the classifier part in real-time, and the final classification accuracy of the ensemble learning scheme could be improved.

- **Alternative combination rules in Dempster-Shafer Theory.** Although Dempster's rule used to combine the BBAs is very popular and widely applied, Zadeh [156] has figured out that Dempster's rule can provide counter-intuitive decisions for an inappropriate design of BBA. Accordingly, many other combination rules have been proposed after that, e.g. Yager's rule [157], Zhang's rule [158] and the cautious rule [159]. They have been applied and analyzed in a wide range of applications. Fei *et al.* [49] have applied the cautious rule to the segmentation of SAS imagery. According to their analysis, the cautious rule is in certain circumstances superior to Dempster's rule. Hence, the introduction of alternative combination rules may provide a promising perspective for the Dempster-Shafer theory based ensemble learning scheme.

List of Abbreviations

ADAC	Automatic Detection and Automatic Classification
ADI	Amount of Duplicate Information
ATR	Automatic Target Recognition
AUV	Autonomous Underwater Vehicle
BBA	Basic Belief Assignment
CMI	Conditional Mutual Information
COOC	co-occurrence matrix
CRM	Composite Relevance Measure
DC	Direct Current
DEM	Diffused Expectation-Maximization
DoC	Degree of Curving
DST	Dempster-Shafer Theory
EM	Expectation Maximization algorithm
E-DS-M	Expectation Maximization algorithm with Dempster-Shafer clustering as intermediate step
GLRL	Gray Level Run Length matrix
ICM	Iterated Conditional Mode
KNN	k-Nearest Neighbor algorithm
KNND	k-Nearest Neighbor algorithm assisted by Dempster-Shafer theory
MAP	Maximum A Posteriori
MCRM-SFS	Maximum Composite Relevance Measure using a Sequential Forward Search
MCRM-SFSA	MCRM-SFS employing J_a
MCRM-SFSG	MCRM-SFS employing J_g

MD	Manhattan Distance
MI	Mutual Information
MIFS	Mutual Information based Feature Selection
MIFS-U	Mutual Information based Feature Selection under Uniform information distribution
MLO	Mine-Like Object
MRF	Markov Random Field
mRMR	minimum Redundancy Maximum Relevance feature selection
mRW	modified Relief Weight
NAS	Non-synthetic Aperture Sonar
PCA	Principal Component Analysis
PNN	Probabilistic Neural Network
RELFSS	Feature Subset Selection based on Relevance
ROI	Regions of Interest
SAS	Synthetic Aperture Sonar
SE	Shannon Entropy
SFS	Sequential Forward Search
SVMG	Supported Vector Machine using a Gaussian kernel
VI	Variation of Information

List of Symbols

a, $\mathfrak{a}_i$	the four parameters controlling the shape of the distribution in Pearson system, $i = 0, 1, 2$
A	the area of the region of interest
$\mathbb{A}$	the region of optimal parameters associated with J_a
$\tilde{\mathbb{A}}$	the complete parameter space associated with J_a
ASD	the angular second moment
$\mathfrak{b}$	the basic belief assignment (BBA) in image segmentation
$\mathfrak{b}_\mathfrak{e}$	the BBA associated with classifier $\mathfrak{e}$ in ensemble learning
$\mathfrak{b}_\oplus$	the combined BBA in ensemble learning
$\mathcal{B}$	co-occurrence matrix
$\bar{\mathcal{B}}$	the normalized version of $\mathcal{B}$
$c^{(m)}$	the m-th realization of C
$\mathfrak{c}_\mathfrak{e}$	the classifier part of classifier $\mathfrak{e}$
C	the random variable denoting the class index
$\mathcal{C}$	the set containing all the possible values of class indices
Comp	the compactness of a given contour
Correlation	the feature *Correlation* obtained by co-occurrence matrix
d	the distance measure proposed in the feature space $\mathbb{F}$
d_M	the Manhattan distance
$\mathfrak{o}_\mathfrak{e}$	the object part associated with classifier $\mathfrak{e}$
DE	the feature difference entropy obtained by co-occurrence matrix
$\mathrm{Den}_\mathrm{ring}$	ring projection condensity
DoC	degree of curving
DV	the feature Difference Variance obtained by co-occurrence matrix
$\mathfrak{e}$	an element of set $\mathfrak{E}$
$\mathfrak{E}$	the set of classifiers applied in this thesis
Ecc	the eccentricity of a given region
Entropy	the feature *Entropy* obtained by co-occurrence matrix
Extent	The feature *Extent* of a given region
$\mathfrak{f}$	a classifier combination function
f_ring, f_radius	the ring and radius projection function
$\bar{f}_\mathrm{ring}$, $\bar{f}_\mathrm{radius}$	the normalized ring and radius projection function
F_i	the i-th distribution type in Pearson system

$\boldsymbol{\mathcal{F}}$	the set of distributions contained in Pearson system
$\mathbb{F}$	the feature space induced by $\mathbf{S}$
$\mathcal{G}_i$	the seven Hu's invariant moments for $i = 1, ..., 7$
G	the sufficiency associated CRM
$\mathbb{G}$	the region of optimal parameters associated with J_g
$\tilde{\mathbb{G}}$	the complete parameter space associated with J_g
H	the Shannon entropy
$\boldsymbol{\mathcal{H}}$	the GLRL matrix
i_{R}	the Rand index
I	the mutual information
I_{VI}	the variation of information
$\boldsymbol{\mathcal{I}}$	the set of pixel indices
Inertia	the feature *Inertia* obtained by co-occurrence matrix
J_a	the CRM with weighted arithmetic average in the method MCRM-SFSA
J_g	the CRM with weighted geometric average in the method MCRM-SFSG
J_{MIFS}	the relevance measure in the method MIFS
$J_{\mathrm{MIFS\text{-}U}}$	the relevance measure in the method MIFS-U
J_{mRMR}	the relevance measure in the method mRMR
J_{RELFSS}	the relevance measure in the method RELFSS
$\mathfrak{K}$	the normalizing factor involved in Dempster's rule
l_i	the label of i-th pixel involved in image segmentation
$l_{\mathrm{minor}}, l_{\mathrm{major}}$	the lengths of principal axes
$\boldsymbol{\mathfrak{L}}_i$	the vector containing the labels of pixels in $\boldsymbol{\mathcal{N}}_i$
$\boldsymbol{\mathcal{L}}$	1.) the set of all the possible states of the pixel labels
	2.) the frame of discernment of hypotheses about pixel labels
$\mathbf{l}$	the label image
M	the number of detected objects in the database
$n_{\boldsymbol{\mathcal{H}}}$	the run length
m_{correct}	the number of correctly identified objects
M_l	the cardinality of $\boldsymbol{\mathcal{L}}$
N_g	the cardinality of set $\mathbb{U}$
$N_{\boldsymbol{\mathcal{H}}}$	the cardinality of $\mathbb{N}_{\boldsymbol{\mathcal{H}}}$
N_O	the cardinality of set $\mathbf{O}$
N_u	the number of pixels in the image
N_S	the cardinality of set $\mathbf{S}$

N_x, N_y	the side lengths of matrix $\tilde{\mathbf{u}}$
$N_{u,j}$	the cardinality of set $\boldsymbol{\mathcal{S}}_j$
$\boldsymbol{\mathcal{N}}_i$	the neighborhood of i-th pixel
$\mathbb{N}_{\boldsymbol{\mathcal{H}}}$	the set of different run lengths
$\mathbf{O}$	the complete set of features
P_{con}	the perimeter of a given contour
Promenance	the feature *Promenance* obtained by co-occurrence matrix
$q_{a,\mathfrak{e}}, q_{g,\mathfrak{e}}$	the classification performance gain indicator for classifier $\mathfrak{e}$
$Q_{a,\text{en}}, Q_{g,\text{en}}$	the ensemble learning performance gain indicator
$\mathbf{r}_i$	the indicator vector of pixel i
$r_{i,j}$	the j-th element of the indicator vector i
$\mathfrak{r}_s, \mathfrak{r}_r$	the substitution rate and rejection rate
R_{area}	the area ratio
R_{axis}	the axis ratio
$R_{c,1}$, $R_{c,2}$	the circularity ratios
R_{va}	the circle variance
RF_1	the feature short runs emphasis obtained by $\boldsymbol{\mathcal{H}}$
RF_2	the feature long runs emphasis obtained by $\boldsymbol{\mathcal{H}}$
RF_3	the feature gray level nonuniformity obtained by $\boldsymbol{\mathcal{H}}$
RF_4	the feature run length nonuniformity obtained by $\boldsymbol{\mathcal{H}}$
RF_5	the feature run percentage obtained by $\boldsymbol{\mathcal{H}}$
$\mathfrak{s}_1$	the square of the skewness
$\mathfrak{s}_2$	the kurtosis
$\mathsf{s}_{a,\mathfrak{e}}$,	the standard deviation of $\rho_{a,\mathfrak{e}}(\boldsymbol{\Gamma}_a)$ over $\mathbb{A}$
$\mathsf{s}_{g,\mathfrak{e}}$,	the standard deviation of $\rho_{g,\mathfrak{e}}(\boldsymbol{\Gamma}_g)$ over $\mathbb{G}$
$\mathsf{s}_{a,\text{en}}$,	the standard deviation of $\rho_{a,\text{en}}(\boldsymbol{\Gamma}_a)$ over $\mathbb{A}$
$\mathsf{s}_{g,\text{en}}$,	the standard deviation of $\rho_{g,\text{en}}(\boldsymbol{\Gamma}_g)$ over $\mathbb{G}$
SA	the feature sum average obtained by co-occurrence matrix
SE	the feature sum entropy obtained by co-occurrence matrix
Shade	the feature *Shade* obtained by co-occurrence matrix
Sol	the solidity of a given contour
$\mathcal{S}$	a segmentation of image
SV	the feature sum variance obtained by co-occurrence matrix
$\mathbf{S}$	the set of the selected features (the feature selection), i.e. a subset of $\mathbf{O}$
$\boldsymbol{\mathcal{S}}_j$	j-th group of image pixels

$\mathcal{T}$	the function used to select features
u_i	the intensity of the i-th pixel in the observed image
$\tilde{u}$	the integer pixel intensity value after transformation
$u_{\boldsymbol{\mathcal{H}}}$	an element out of set $\mathbb{U}$
$\tilde{\mathbf{u}}$	the 2D image of dimension of $N_x \times N_y$
$\mathbf{u}$	the array version of observed image
$\mathfrak{u}_i$	the intensity of the i-th pixel in the unknown noise-free image
$\mathfrak{U}$	the set of all possible states of $\mathfrak{u}_i$
$\mathbb{U}$	the set of all possible states of the pixel values in $\tilde{\mathbf{u}}$
v_η	the support of pixel η
$\mathcal{V}$	the roughness of a given contour
$w_{i,j}$	the probability of $r_{i,j}$ equals to 1
$\mathfrak{w}$	the distance difference associated with an individual object in mRW calculation
W	the modified Relief weight
(x_*, y_*)	the centroid of a region
$\mathcal{X}_n$	the n-th feature
$\mathfrak{y}_\mathfrak{e}$	the output support of classifier $\mathfrak{e}$
$\mathfrak{z}$	the complete data
β_i	the parameter controlling the cliques, $i = 1, ..., 4$
ϵ_i	the additive noise in i-th pixel
$\varepsilon_{\text{ring}}$	the ring projection skewness
$\varepsilon_{\text{radius}}$	the radius projection skewness
$\delta_{\text{Kronecker}}$	the Kronecker delta function
$\gamma_{a,W}, \gamma_{a,H}$	the parameters involved in J_a
$\gamma_{g,W}, \gamma_{g,H}$	the parameters involved in J_g
$\boldsymbol{\Gamma}_a, \boldsymbol{\Gamma}_g$	the parameter vectors associated with J_a and J_g, respectively
κ_{mean}	the absolute curvature mean value of a given contour
$\varkappa$	the parameter specifying the belief in the assumption of pairwise class-conditional independence
γ_1, γ_2	the parameters associated with ϑ_η and v_η, respectively
$\pi_{i,j}$	the mixing coefficient involved in the mixture model
ρ	the identification rate
$\boldsymbol{\chi}^{(m)}$	the feature vector of m-th object

$\boldsymbol{\chi}_S^{(m)}$	the m-th point in the subspace induced by $\mathbf{S}$
$\chi_{n,m}$	the n-th element of feature vector $\boldsymbol{\chi}^{(m)}$
$\boldsymbol{\Pi}$	the vector containing all the mixing coefficients in the mixture model
$\boldsymbol{\psi}_j$	the parameters required for the distribution when $l_i = j$
$\boldsymbol{\Psi}$	the parameter vector containing the parameters of all the component distributions in the mixture model
$\boldsymbol{\Phi}$	the parameter vector containing all the parameters involved in the mixture model
Υ	the energy function
ϱ_{LF}	the low frequency density
$\varepsilon_{\mathrm{DFT}}$	the Fourier coefficient skewness
ϑ_η	the quality of pixel η
μ_j	the mean value of the pixel intensities associated with j-th class
ν_i	the median of the pixel intensities in $\boldsymbol{\mathcal{N}}_i$
μ_{ring}	the radius projection mean value
$\zeta_{n,j}$	the n-th central moment of j-th class

Bibliography

[1] B. Bhanu, "Automatic target recognition: State of the art survey," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 22, no. 4, pp. 364 –379, 1986.

[2] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys (CSUR)*, vol. 35, no. 4, pp. 399458, 2003.

[3] P.J. Phillips, Hyeonjoon M., S. A. Rizvi, and P.J. Rauss, "The feret evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 109–1104, 2000.

[4] A. Zimek, F. Buchwald, E. Frank, and S. Kramer, "A study of hierarchical and flat classification of proteins," *Transactions on Computational Biology and Bioinformatics, IEEE/ACM*, vol. 7, no. 3, pp. 563571, 2010.

[5] S.G. Johnson and A. Deaett, "The application of automated recognition techniques to side-scan sonar imagery," *Journal of Oceanic Engineering*, vol. 19, no. 1, pp. 138–144, 1994.

[6] R. Fandos and A. M. Zoubir, "Optimal feature set for automatic detection and classification of underwater objects in SAS images," *IEEE J. Sel. Top. Sign. Proces*, vol. 5, no. 3, pp. 454–468, Jun. 2011.

[7] J. E. Piper, R. Lim, E. I. Thorsos, and K. L. Williams, "Buried sphere detection using a synthetic aperture sonar," *IEEE J. Oceanic Eng.*, vol. 34, no. 4, pp. 485–494, Oct. 2009.

[8] J.E. Piper, K.W. Commander, E.I. Thorsos, and K.L. Williams, "Detection of buried targets using a synthetic aperture sonar," *IEEE J. Oceanic Eng.*, vol. 27, no. 3, pp. 495–504, Jul. 2002.

[9] V. Myers and D.P. Williams, "Adaptive multiview target classification in synthetic aperture sonar images using a partially observable Markov decision process," *IEEE J. Oceanic Eng.*, vol. 37, no. 1, pp. 45–55, Jan. 2012.

[10] C. Debes, J. Hahn, A. M. Zoubir, and M. Amin, "Target discrimination and classification in through-the-wall radar imaging," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4664–4676, 2011.

[11] L. Carin, N. Geng, M. McClure, J. Sichina, and L. Nguyen, "Ultra-wide-band synthetic-aperture radar for mine-field detection," *IEEE Antennas Propag. Mag.*, vol. 41, no. 1, pp. 18–33, Feb. 1999.

[12] Q. Zhao and J. C. Principe, "Support vector machines for SAR automatic target recognition," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, no. 2, pp. 643 –654, Apr. 2001.

[13] K. T. Kim, D. K. Seo, and H. T. Kim, "Efficient radar target recognition using the MUSIC algorithm and invariant features," *IEEE Trans. Antennas Propag.*, vol. 50, no. 3, pp. 325 –337, Mar. 2002.

[14] L. Du, H. Liu, and Z. Bao, "Radar automatic target recognition based on complex high-resolution range profiles," *Proc. Int. Conf. Radar 2006. CIE '06*, pp. 1–5, Oct. 2006.

[15] L. J. Cutrona, "Additional characteristics of synthetic aperture sonar systems and a further comparison with non-synthetic aperture sonar systems," *J. Acoust. Soc. Am.*, vol. 61, no. 5, pp. 1213–1217, 1977.

[16] L. Henriksen, "Real-time underwater object detection based on an electrically scanned high-resolution sonar," *Proc. Symp. Autonomous Underwater Vehicle Technology 1994, AUV '94*, pp. 99 –104, Jul. 1994.

[17] M. R. Azimi-Sadjadi, D. Yao, Q. Huang, and G. J. Dobeck, "Underwater target classification using wavelet packets and neural networks," *IEEE Trans. Neural Netw.*, vol. 11, no. 3, pp. 784–794, May 2000.

[18] S. Reed, Y. Petillot, and J. Bell, "An automatic approach to the detection and extraction of mine features in sidescan sonar," *IEEE J. Ocean. Eng.*, vol. 28, no. 1, pp. 90–105, 2003.

[19] A. D. Matthews, T. C. Montgomery, D. A. Cook, J. W. Oeschger, and J. S. Stroud, "12.75-inch synthetic aperture sonar (SAS), high resolution and automatic target recognition," *OCEANS'06*, pp. 1–7, 2006.

[20] C. Rao, K. Mukherjee, S. Gupta, A. Ray, and S. Phoha, "Underwater mine detection using symbolic pattern analysis of sidescan sonar images," *Proc. American Control Conference (ACC '09)*, pp. 5416 –5421, Jun. 2009.

[21] M. F. Doherty, J. G. Landowski, P. F. Maynard, G. T. Uber, D. W. Fries, and F. H. Maltz, "Side scan sonar object classification algorithms," *Proc. 6th Int. Symp. on Unmanned Untethered Submersible Technology*, p. 417424, 1989.

[22] G. J. Dobeck, J. C. Hyland, and L. Smedley, "Automated detection and classification of sea mines in sonar imagery," *Proc. SPIE Int. Soc. Opt*, vol. 3079, pp. 90–110, Jul. 1997.

[23] E. Coiras, P.-Y. Mignotte, Y. Petillot, J. Bell, and K. Lebart, "Supervised target detection and classification by training on augmented reality data," *IET Radar, Sonar & Navigation*, vol. 1, no. 1, pp. 83–90, 2007.

[24] Y. Petillot, S. Reed, and E. Coiras, "An augmented reality solution for evaluating underwater sonar mcm systems," *Proc. 7th Int. Symp. on Technology and the Mine Problem*, vol. 1, no. 1, pp. 83–90, 2007.

[25] T. Fei and D. Kraus, "An expectation-maximization approach assisted by Dempster-Shafer theory and its application to sonar image segmentation," *Proc. of IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP 2012)*, pp. 1161–1164, 2012.

[26] M. A. Shackleton and W. J. Welsh, "Classification of facial features for recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition*, pp. 573 –579, Jun. 1991.

[27] X. Huang, L. Zhang, and P. Li, "Classification and extraction of spatial features in urban areas using high-resolution multispectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 2, pp. 260–264, Apr. 2007.

[28] Q. Li and D. W. Tufts, "Principal feature classification," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 155–160, Jan. 1997.

[29] K. Huang and S. Aviyente, "Wavelet feature selection for image classification," *IEEE Trans. Image Process.*, vol. 17, no. 9, pp. 1709–1720, Sep. 2008.

[30] L. Liu and P. Fieguth, "Texture classification from random features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 574–586, Mar. 2012.

[31] S. Li, M. C. Lee, and C. M. Pun, "Complex Zernike moments features for shape-based image retrieval," *IEEE Trans. Syst. Man, Cybern.*, vol. 39, no. 1, pp. 227–237, Jan. 2009.

[32] Z. Xu, Hong R. Wu, X. Yu, K. Horadam, and B. Qiu, "Robust shape-feature-vector-based face recognition system," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 12, pp. 3781–3791, Dec. 2011.

[33] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006, ch. 1.4.

[34] K. R. Muller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181 –201, Mar. 2001.

[35] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 381–396, Mar. 2002.

[36] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 988–999, Sep. 1999.

[37] T. Pun, "A new method for grey-level picture thresholding using the entropy of the histogram," *Signal Process.*, vol. 2, pp. 223237, 1980.

[38] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, "A new method for grey-level picture thresholding using the entropy of the histogram," *Comput. Vis. Graph. Image Process.*, vol. 29, pp. 273285, 1985.

[39] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Y. C. Chen, "A survey of thresholding techniques," *Comput. Vis. Graph. Image Process.*, vol. 41, pp. 233260, 1998.

[40] S. U. Lee, S. Y. Chung, and R. H. Park, "A comparative performance study of several global thresholding techniques for segmentation," *Comput. Vis. Graph. Image Process.*, vol. 52, pp. 171190, 1990.

[41] P. K. Sahoo, D. W. Slaaf, and T.A. Albert, "Threshold selection using a minimal histogram entropy difference," *Opt. Eng.*, vol. 36, pp. 19761981, 1997.

[42] O.J. Tobias and R. Seara, "Image segmentation by histogram thresholding using fuzzy sets," *IEEE Trans. Image Process.*, vol. 11, no. 12, pp. 1457 – 1465, 2002.

[43] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *INT. J. Comput. Vision*, vol. 23, pp. 321–331, 1988.

[44] J. Zhang, J. W. Modestino, and D. A. Langan, "Maximum-likelihood parameter estimation for unsupervised stochastic model-based image segmentation," *IEEE Trans. Image Process.*, vol. 3, no. 4, pp. 404–420, 1994.

[45] Y. Delignone, A. Marzouki, and W. Pieczynski, "Estimation of generalized mixtures and its application in image segmentation," *IEEE Trans. Image Process.*, vol. 6, no. 10, pp. 1364–1375, Oct. 1997.

[46] S. Sanjay-Gopal and T. J. Hebert, "Bayesian pixel classification using spatially variant finite mixtures and the generalized EM algorithm," *IEEE Trans. Image Process.*, vol. 7, no. 7, pp. 1014–1028, 1998.

[47] G. Boccignone, V. Caggiano, P. Napoletano, and M. Ferraro, "Image segmentation via multiresolution diffused expectation-maximisation," *Proc. IEEE ICIP '05*, vol. 1, pp. 289–292, 2005.

[48] M. Mignotte, C. Collet, P. Perez, and P. Bouthemy, "Sonar image segmentation using an unsupervised hierarchical MRF model," *IEEE Trans. Image Process.*, vol. 9, pp. 1216–1231, 1998.

[49] T. Fei and D. Kraus, "An evidence theory supported expectation-maximization approach for sonar image segmentation," *Proc. of IEEE Int. Conf. Multi-Conference on On Communication and Signal Processing (SSD '12 -CSP)*, pp. 1–6, 2012.

[50] M. Yang, K. Kpalma, and J. Ronsin, "A survey of shape feature extraction techniques," *Pattern Recognition, Peng-Yeng Yin (Ed.)*, pp. 43–90, 2008.

[51] A. Castellano and B. Gray, "Autonomous interpretation of side scan sonar returns," *Proc. Symp. Autonomous Underwater Vehicle Technology*, pp. 248–253, 1990.

[52] I. Qiudu, J. Malkasse, G. Burel, and P. Vilbe, "Mine classification using a hybrid set of descriptors," *Proc. Oceans Conference*, vol. 1, pp. 291–297, 2000.

[53] J. C. Delvigne, "Shadow classification using neural networks," *4th Undersea Defence Conference*, pp. 214–221, 1992.

[54] Matthew Turk and Alex Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[55] B. Schölkopf, A. Smola, and K. R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, pp. 1299–1319, 1998.

[56] M. Dash and H. Liu, "Feature selection for classification," *Intell. Data Anal.*, vol. 1, pp. 131–156, 1997.

[57] A. G. K. Janecek and W. N. Gansterer, "On the relationship between feature selection and classification accuracy," *JMLR: Workshop and Conference Proceedings*, pp. 90–105, 2008.

[58] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of Relieff and RRelieff," *Mach. Learn.*, vol. 53, no. 1-2, pp. 23–69, Oct. 2003.

[59] L. C. Molina, L. Belanche, and A. Nebot, "Feature selection algorithms: A survey and experimental evaluation," *Proc. IEEE Int. Conf. Data Mining*, pp. 306–313, 2002.

[60] D. F. Specht, "Probabilistic neural networks," *Neural Netw.*, vol. 3, no. 1, pp. 109–118, 1990.

[61] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[62] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: a review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 4–37, 2000.

[63] T. Denoeux, "A k-nearest neighbor classification rule based on Dempster-Shafer theory," *IEEE Trans. Syst. Man, Cybern.*, vol. 25, no. 5, pp. 804–813, 1995.

[64] D. D. Nguyen, K. Matsumoto, Y. Takishima, and K. Hashimoto, "Condensed vector machines: Learning fast machine for large data," *IEEE Trans. Neural Netw.*, vol. 21, no. 12, pp. 1903 – 1914, 2010.

[65] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons., 2nd edition, 2001, ch. 4.6.

[66] C. Y. Suen, C. Nadal, T. A. Mai, R. Legault, and L. Lam, "Recognition of totally unconstrained handwritten numerals based on the concept of multiple experts," *Proc. Int. Workshop on Frontiers in Handwriting Recognition*, pp. 131–143, Apr. 1990.

[67] C. Nadal, R. Legault, and C. Y. Suen, "Complementary algorithms for the recognition of totally unconstrained handwritten numerals," *Proc. 10th Int. Conf. Pattern Recog.*, vol. A, pp. 434–449, Jun. 1990.

[68] J. J. Hull, A Commike, and T. K. Ho, "Multiple algorithms for handwritten character recognition," *Proc. Int. Workshop Frontiers in Handwriting Recognition*, pp. 117–129, Apr. 1990.

[69] S. Hashem and B. Schmeiser, "Improving model accuracy using optimal linear combinations of trained neural networks," *IEEE Trans. Neural Netw.*, vol. 6, no. 3, pp. 792–794, 1995.

[70] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Trans. Syst. Man, Cybern.*, vol. 22, no. 3, pp. 418–435, 1992.

[71] J. Kittler, M. Hatef, and R. P. W. Duin, "Combining classifiers," *Proc. 13th Intl Conf. Pattern Recognition*, vol. 2, pp. 897–901, 1996.

[72] J. Kittler, S. A. Hojjatoleslami, and T. Windeatt, "Weighting factors in multiple expert fusion," *Proc. British Machine Vision Conf.*, pp. 41–50, 1997.

[73] G. Rogova, "Combining the results of several neural network classifiers," *Neural Netw.*, vol. 7, no. 5, pp. 777–781, 1994.

[74] Jun Sun, Kaizhu Huang, Y. Hotta, K. Fujimoto, and S. Naoi, "Degraded character recognition by complementary classifiers combination," *Proc. 9th ICDAR*, vol. 2, pp. 579 –583, sept. 2007.

[75] H. El-Shishini, M.S. Abdel-Mottaleb, M. El-Raey, and A. Shoukry, "A multistage algorithm for fast classification of patterns," *Pattern Recognition Letters*, vol. 10, no. 4, pp. 211–215, 1989.

[76] M.C. Fairhurst and H.M.S. Abdel Wahab, "An interactive two-level architecture for a memory network pattern classifier," *Pattern Recognition Letters*, vol. 11, no. 8, pp. 537–540, 1990.

[77] P. Pudil, J. Novovicova, S. Blaha, and J. Kittler, "Multistage pattern recognition with reject option," *Proc. 11th IAPR Int. Conf. Pattern Recognition, Conf. B: Pattern Recognition Methodology and Systems*, vol. 2, pp. 92–95, 1992.

[78] D. A. Denisov and A. K. Dudkin, "Model-based chromosome recognition via hypotheses construction/verification," *Pattern Recognition Letters*, vol. 15, no. 3, pp. 299–307, 1994.

[79] C. H. Tung, H. J. Lee, and J. Y. Tsai, "Multi-stage pre-candidate selection in handwritten Chinese character recognition systems," *Pattern Recognition*, vol. 27, no. 8, pp. 1093–1102, 1994.

[80] J. Y. Zhou and T. Pavlidis, "Discrimination of characters by a multi-stage recognition process," *Pattern Recognition*, vol. 27, no. 11, pp. 1539–1549, 1994.

[81] J. Franke and E. Mandler, "A comparison of two approaches for combining the votes of cooperating classifiers," *Proc. 11th IAPR Intl Conf. Pattern Recognition, Conf. B: Pattern Recognition Methodology and Systems*, vol. 2, pp. 611–614, 1992.

[82] F. Kimura and M. Shridhar, "Handwritten numerical recognition based on multiple algorithms," *Pattern Recognition*, vol. 24, no. 10, pp. 969–983, 1991.

[83] S. C. Bagui and N. R. Pal, "A multistage generalization of the rank nearest neighbor classification rule," *Pattern Recognition Letters*, vol. 16, no. 6, pp. 601–614, 1995.

[84] T. K. Ho, J. J. Hull, and S. N. Srihari, "Decision combination in multiple classifier systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 1, pp. 66–75, 1994.

[85] B. Zhang and S. N. Srihari, "Class-wise multi-classifier combination based on Dempster-Shafer theory," *Proc. 7th Int. Conf. Control, Automation, Robotics and Vision, ICARV 2002*, vol. 2, pp. 698–703, 2002.

[86] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," *Proc. AAAI-92*, pp. 129–134, 1992.

[87] N. Chen, W. Lu, J. Yang, and G. Li, *Support vector machine in chemistry*, World Scientific, 2004.

[88] E. W. Forgy, "Cluster analysis of multivariate data: efficiency vs interpretability of classifications," *Biometrics*, vol. 21, pp. 768–769, 1965.

[89] M. Lianantonakis and Y. R. Petillot, "Sidescan sonar segmentation using active contours and level set methods," *Oceans 2005 - Europe*, vol. 1, pp. 719–724, 2005.

[90] J. Besag, "On the statistical analysis of dirty pictures," *J. R. Stat. Soc. Series B (Methodological)*, vol. 48, no. 3, pp. 259–302, 1986.

[91] O. Demirkaya, M. H. Asyali, and P. Sahoo, *Image processing with MATLAB : applications in medicine and biology*, CRC Press, 6000 Broken Sound Parkway NW, Suite 300, Boca Raton, FL 33487-2742, 2009.

[92] R. Kindermann and J. L. Snell, *Markov random fields and their applications*, vol. 1 of *Contemporary mathematics*, American Mathematical Society, 1980.

[93] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Stat. Soc. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[94] J. Weickert, "Theoretical foundations of anisotropic diffusion in image processing," *Computing, Suppl.*, vol. 11, pp. 221–236, 1996.

[95] S. B. Chaabane, M. Sayadi, F. Fnaiech, and E. Brassart, "Color image segmentation based on Dempster-Shafer evidence theory," *Proc. IEEE MELECON '08.*, pp. 862–866, 2008.

[96] S. B. Chaabane, F. Fnaiech, M. Sayadi, and E. Brassart, "Relevance of the Dempster-Shafer evidence theory for image segmentation," *Proc. IEEE SCS '09 (3rd)*, pp. 1–4, 2009.

[97] S. B. Chaabane, M. Sayadi, F. Fnaiech, and E. Brassart, "Dempster-Shafer evidence theory for image segmentation: application in cells images," *IJICT*, vol. 5, no. 2, pp. 126–132, 2009.

[98] Y. Neng-Hai and Y. Yong, "Multiple level parallel decision fusion model with distributed sensors based on Dempster-Shafer evidence theory," *Proc. Int. Conf. Mach. Learn. Cyber.*, vol. 5, pp. 3104–3108, 2003.

[99] S. Salicone, *Measurement Uncertainty: An Approach via the Mathematical Theoy of Evidence*, Springer, 2006.

[100] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*, vol. 1, Wiley-Interscience, 2nd edition, 1994.

[101] G. J. Woodward, "Approximations of Pearson Type IV tail probabilities," *Journal of the American Statistical Association (JASA)*, vol. 71, pp. 513–514, 1976.

[102] J. Heinrich, "A guide to the Pearson Type IV distribution," Tech. Rep., University of Pennsylvania, 2004.

[103] M. Gürtler, J. P. Kreiss, and R. Rauh, "A non-stationary approach for financial returns with nonparametric heteroscedasticity," Tech. Rep., Institut für Finanzwirtschaft, Technische Universität Braunschweig, Sep. 2009.

[104] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 6, pp. 721–741, 1984.

[105] J. Besag, "Spatial interaction and the statistical analysis of lattice systems," *J. R. Stat. Soc. Series B (Methodological)*, vol. 36, no. 2, pp. 192–236, 1974.

[106] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*, Series in Statistics. Springer-Verlag, 2. edition, 1985.

[107] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *The Annals of Mathematical Statistics*, vol. 38, no. 2, pp. 325–339, 1967.

[108] G. Shafer, *A Mathematical Theory of Evidence*, Princeton University Press, 1976.

[109] P. Smets, "Constructing the pignistic probability function in a context of uncertainty," *Proc. 5th Ann. Conf. Uncertainty in Artificial Intel.*, pp. 29–39, 1989.

[110] P. Smets and R. Kennes, "The transferable belief model," *Artif. Intell.*, vol. 66, pp. 191–243, 1994.

[111] L. Liu and R. R Yager, "Classic works of the Dempster-Shafer theory of belief functions: A introduction," *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pp. 1–34, 2008.

[112] A. R. Webb, "Gamma mixture models for target recognition," *Pattern Recognition*, vol. 33, no. 12, pp. 2045–2054, Dec. 2000.

[113] R. Fandos and A. M. Zoubir, "Enhanced initialization scheme for a three region Markovian segmentation algorithm and its application to SAS images," *Proc. 10th Europ. Conf. Underwater Acoustics (ECUA)*, vol. 3, pp. 1323–1331, 2010.

[114] M. Meilă, "Comparing clustering by the variation of information," *Proc. 6th Ann.Conf. Compt. Learning Thoery (COLT)*, pp. 173–187, 2003.

[115] D. Tang, F. S. Henyey, B. T. Hefner, and P. A. Traykovski, "Simulating realistic-looking sediment ripple fields," *IEEE J. Oceanic Eng.*, vol. 34, no. 4, pp. 444–450, Oct. 2009.

[116] I. Guyon and L. A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research (JMLR)*, vol. 3, pp. 1157–1182, Mar. 2003.

[117] M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory,*, vol. 8, no. 2, pp. 179–187, 1962.

[118] R. M. Haralick, K. Shanmugam, and Its'Hak Dinstein, "Textural features for image classification," *IEEE Trans. Syst. Man, Cybern.*, vol. 3, no. 6, pp. 610–621, 1973.

[119] J. S. Weszka, C. R. Dyer, and A. Rosenfeld, "A comparative study of texture measures for terrain classification," *IEEE Trans. Syst. Man, Cybern.*, vol. 6, no. 4, pp. 269 –285, Apr. 1976.

[120] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Inc., 1989, ch. 9.

[121] M. Peura and J. Iivarinen, "Efficiency of simple shape descriptors," *Proc. 3rd Int. Workshop on Visual Form (IWVF3)*, pp. 28–30, 1997.

[122] Y. Y. Tang, *Wavelet Theory Approach to Pattern Recognition*, 74. World Scientific Publishing Co. Pte. Ltd., 2nd edition, 2009, ch. 9.

[123] R. F. Walker, P. Jackway, and I. D. Longstaff, "Improving co-occurrence matrix feature discrimination," *Proc. of DICTA95, 3rd International Conference on Digital Image Computing: Techniques and Applications*, pp. 643–648, 1995.

[124] M. M. Galloway, "Texture analysis using gray level run lengths," *Computer Graphics and Image Processing*, vol. 4, no. 2, pp. 172–179, Jun. 1975.

[125] H. H. Yang and J. Moody, "Feature selection based on joint mutual information," *Proc. Int. ICSC Symp. Advances in Intelligent Data Analysis*, pp. 22–25, 1999.

[126] D. A. Bell and H. Wang, "A formalism for relevance and its application in feature subset selection," *Mach. Learn.*, vol. 41, no. 2, pp. 175–195, Nov. 2000.

[127] G. Brown, A. Pocock, M. J. Zhao, and M. Lujn, "Conditional likelihood maximisation: A unifying framework for information theoretic feature selection," *Journal of Machine Learning Research (JMLR)*, vol. 13, pp. 27–66, 2012.

[128] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Trans. Neural Netw.*, vol. 5, pp. 537–550, 1994.

[129] N. Kwak and C. H. Choi, "Input feature selection for classification problems," *IEEE Trans. Neural Netw.*, vol. 13, no. 1, pp. 143–159, Jan. 2002.

[130] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, 2005.

[131] T. Fei, D. Kraus, and P. Berkel, "A new idea on feature selection and its application to the underwater object recognition," *Proc. 11th Europ. Conf. Underwater Acoustics (ECUA 2012)*, pp. 52–59, 2012.

[132] T. Fei, D. Kraus, and A. M. Zoubir, "A hybrid relevance measure for feature selection and its application to underwater objects recognition," *Proc. IEEE ICIP '12*, pp. 97–100, 2012.

[133] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons. Inc., 1991.

[134] J. P. Vert, K. Tsuda, and B. Schölkopf, "A primer on kernel methods," *Kernel Methods in Computational Biology*, pp. 35–70, 2004.

[135] S. Canu, Y. Grandvalet, V. Guigue, and A. Rakotomamonjy, "SVM and kernel methods Matlab Toolbox," *Perception Systemes et Information, INSA de Rouen, Rouen, France*, 2005.

[136] J. E. Goin, "Classification bias of the k-nearest neighbor algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 3, pp. 379–381, 1984.

[137] J. M. Keller, M. R. Gray, and J. A. J. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE Trans. Syst. Man, Cybern.*, vol. 15, no. 4, pp. 580–584, 1985.

[138] R.L. Streit and T.E. Luginbuhl, "Maximum likelihood training of probabilistic neural networks," *IEEE Transactions on Neural Networks, DOI - 10.1109/72.317728*, vol. 5, no. 5, pp. 764–783, 1994.

[139] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Trans. Syst. Man, Cybern.*, vol. 30, no. 4, pp. 451–462, 2000.

[140] P.-Y. Mignotte, E. Coiras, H. Rohou, Y. Petillot, J. Bell, and K. Lebart., "Adaptive fusion framework based on augmented reality training," *IET Radar Sonar Navig.*, vol. 2, no. 2, pp. 146–154, 2008.

[141] J. Fawcett, V. Myers, D. Hopkins, A. Crawford, M. Couillard, and B. Zerr, "Multiaspect classification of sidescan sonar images, four different approaches to fusing single-aspect information," *IEEE J. Ocean. Eng.*, vol. 35, no. 4, pp. 863–876, 2010.

[142] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Networks*, vol. 13, no. 2, pp. 415–425, Mar. 2002.

[143] Ludmila I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, John Wiley & Sons, Inc, 2004.

[144] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, 1990.

[145] M. Vetterli and C. Herley, "Wavelets and filter banks: theory and design," *IEEE Trans. Signal Process.*, vol. 40, no. 9, pp. 2207–2232, Sep 1992.

[146] Carlos Bazan and Peter Blomgren, *Image Smoothing and Edge Detection by Nonlinear Diffusion and Bilateral Filter*, Ph.D. thesis, San Diego State University, 2008.

[147] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," *Proc. IEEE CVPR 2005*, vol. 2, pp. 60–65, 2005.

[148] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," *Proc. SPIE Electronic Imaging*, vol. 6064, pp. 606414–1 – 606414–12, 2006.

[149] Hong Shi, Hui Wang, Chunhui Zhao, and Zhengyan Shen, "Sonar image preprocessing based on morphological wavelet transform," *Proc. ICIA 2010*, pp. 1113 – 1117.

[150] I. Mandhouj, F. Maussang, and H. Solaiman, B. dan Amiri, "Sonar image preprocessing method based on homomorphic filtering," *Oceans 2012*, pp. 1–5, 2012.

[151] O. Lopera, R. Heremans, A. Pizurica, and Y. Dupont, "Filtering speckle noise in sas images to improve detection and identification of seafloor targets," *Proc. WSS 2010*, pp. 1–5, 2010.

[152] I. Firoiu, C. Nafornita, D. Isar, and A. Isar, "Bayesian hyperanalytic denoising of sonar images," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 6, pp. 1065 – 1069, 2011.

[153] S.D. Stearns, "On selecting features for pattern classifiers," *Third lnternat. Conf. on Pattern Recognition*, pp. 71–75, 1976.

[154] P. Pudil, F.J. Ferri, J. Novovicova, and J. Kittler, "Floating search methods for feature selection with nonmonotonic criterion functions," *Proc. 12th IAPR*, vol. 2, pp. 279 – 283, 1994.

[155] Matthias Ehrgott, *Multicriteria Optimization*, Springer, 2nd. edition, 2005.

[156] L. A. Zadeh, "Reviews of books: A mathematical theory of evidence," *AI Magazine (AAAI)*, vol. 5, pp. 81–83, 1984.

[157] R. R. Yager, "On the Dempster-Shafer framework and new combination rules," *Information Sciences*, vol. 41, pp. 93–138, 1987.

[158] L. Zhang, "Representation, independence, and combination of evidence in the Dempster-Shafer theory," *Advances in the Dempster-Shafer Theory of Evidence*, pp. 51–69, 1994.

[159] T. Denceux, "The cautious rule of combination for belief functions and some extensions," *Proc. 9th Int. Conf. on Information Fusion*, pp. 1–8, 2006.

Curriculum Vitae

Name:	Tai Fei
Date of birth:	27.01.1983
Place of birth:	Suzhou, Jiangsu, P.R. China
Family status:	married

Education

09/2005 - 06/2009	Technische Universität Darmstadt Nachrichten- und Kommunikationstechnik (Diplom-Ingenieur)
09/2001 - 09/2005	Shanghai Maritime University Telekommunikationstechnik (Bachelor of Science)
06/2001	High school degree (Abitur) at No.10 Middle School Suzhou, China

Work experience

03/2014 - today	Development engineer with Hella KGaA Hueck & Co, Lippstadt, Germany
08/2013 - 02/2014	Research associate with Center for Marine Environmental Sciences, University of Bremen
09/2009 - 12/2012	Research associate with Institute of Water-Acoustics, Sonar-Engineering And Signal-Theory Hochschule Bremen
07/2008 - 11/2008	Internship with ALLDOS Eichler GmbH, Germany

Erklärung laut §9 PromO

Ich versichere hiermit, dass ich die vorliegende Dissertation allein und nur unter Verwendung der angegebenen Literatur verfasst habe. Die Arbeit hat bisher noch nicht zu Prüfungszwecken gedient.

Bremen, 01.03.2014,