Spatial and Temporal SAR Image Information Mining

DISSERTATION
zur Erlangung des Grades eines Doktors
der Ingenieurwissenschaften

vorgelegt von
Shiyong Cui, M.Eng.

eingereicht bei der Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen
Siegen 2014

Stand: August 2014

Promotionskommission:

1. Gutachter: Prof. Dr. Otmar Loffeld
2. Gutachter: Prof. Dr. Mihai Datcu

1. Mitglied der Promotionskommision: Prof. Dr. Lorenzo Bruzzone
2. Mitglied der Promotionskommision: Prof. Dr. Cristian Negrescu

Vorsitz des Prüfungskommission: Prof. Dr. Andreas Kolb

Tag der mündlichen Prüfung: 26. August 2014

to my parents, my wife, and our lovely daughter.

Acknowledgements

"It takes a village to raise a child". So it is with this thesis. This thesis would not have been possible without the immense help and support that I received from many friends and colleagues in the Remote Sensing Technology Institute (IMF) of the German Aerospace Center (DLR).

First and foremost, I would like to express my heartfelt gratitude to my supervisor Prof. Dr. Mihai Datcu for his continued encouragement and invaluable suggestions in completing this thesis. Throughout my doctoral research study at DLR, he has always been patient and encouraging in guiding me to the right direction and solving various difficult points. I also appreciate all his contributions of time, ideas, and suggestions to this thesis.

I am also grateful to Prof. Dr. Peter Reinartz for giving me the opportunity to carry out this thesis in IMF. I would like to acknowledge Prof. Dr. Otmar Loffeld and Dr. Holger Nies of ZESS at the University of Siegen for giving me the opportunity to enroll at Siegen University. I would also like to thank Ms. Rebekka Kammler and DAAD for the financial support to carry out this study.

Particularly, I would like to thank my colleague Mr. Gottfried Schwarz and Ms. Theresia Hantel for their kind help in solving various daily difficulties in life during my doctoral research study in DLR. I would also like to thank Ambar Murillo Montes de Oca for her help to correct this thesis.

I would like to thank all other colleagues from IMF: Dr. Daniela Espinoza Molina, Fabio Cian, Dr. Mihai Neghina, Dr. Miguel Angel Veganzones, Jayashree Chadalawada, and Gholamreza Bahmanyar, Dr. Corneliu Octavian Dumitru, Jagmal Singh, Dr. Daniele Cerra, Janja Avbelj, Ke Zhu, Dr. Jiaojiao Tian, Peter Schwind, and Oliver Meynberg. Further, I would like to thank Dr. Pierre Blanchart for his first prototype system of cascade learning, based on which I could continue the development. I would also like to thank other friends, namely Dr. Anca Popescu, Carmen Patrascu, Dr. Corina Vaduva, and Dr. Daniela Faur.

Finally, I would like to thank my family, my parents, my wife, and our lovely daughter. Their love is the driving force of all my inspiration and energy. I owe them everything and wish I could show them just how much I love and appreciate them in the rest of my life.

Zusammenfassung

Der Schwerpunkt dieser Dissertation liegt in der Entwicklung von neuen Methoden zum Information Mining von SAR-Bildern mit hoher räumlicher und zeitlicher Auflösung. Ausgehend von statistischen Modellen schlagen wir zuverlässige Modelle und robuste Methoden zur Parameterschätzung vor und bewerten die statistischen Modelle für verschiedene Klassen von Bildern. Aufbauend auf diesen statistischen Modellen werden MaSSe zur Informationsähnlichkeit bei der Entdeckung von Änderungen in SAR-Bildern sowohl im Ortsraum als auch im Wavelet-Raum angewendet. Zur Beurteilung der Leistungsfähigkeit wurde ein Referenz-Datensatz erzeugt, wo Veränderungen (wie die Statistiken erster, zweiter und höherer Ordnung) simuliert wurden. Dies löst das Problem der fehlenden Referenz-Daten bei der Beurteilung der verschiedenen Methoden und erlaubt eine umfassende Beurteilung von MaSSen zur Informationsähnlichkeit sowohl mit synthetischen als auch mit echten Daten.

Ausgehend von den spezifischen Eigenschaften von SAR-Bildern mit sehr hoher Auflösung werden zwei neue Methoden zur Merkmalsextraktion entwickelt. Die erste ist eine neue Merkmalsextraktions-Methode zur Strukturbeschreibung von SAR-Bildern mit hoher Auflösung, die durch den bekannten verhältnisdiskriminierenden Kantendetektor angeregt wurde. Hier werden Intensitätsverhältnisse in verschiedenen Richtungen innerhalb von lokalen Bildausschnitten angewandt, um die Bag-of-Words-(BoW)-Merkmalsextraktion zu verbessern und um einen lokalen Weber-Deskriptor an SAR-Bilder anzupassen. Die zweite Methode ist eine einfache und dennoch effiziente Methode zur Merkmalsextraktion aus dem Bereich der Bag-of Words (BoW)-Verfahren. Diese Methode beinhaltet zwei wesentliche Neuentwicklungen. Zum einen und was für uns sehr interessant ist, benötigt die Methode keinerlei örtliche Merkmalsextraktion. Stattdessen benutzt sie als einfache Merkmale direkt die Pixelwerte aus einem lokalen Fenster. Zum anderen - und im Gegensatz vielen unüberwachten Lernmethoden für Merkmale - wird ein Zufallswörterbuch für die Quantisierung des Merkmalsraums verwendet. Der Vorteil eines Zufallswörterbuchs ist, dass es zu keinem merklichen Verlust der Klassifizierungsgüte kommt, obwohl der zeitaufwändige Prozess des Erlernens eines Wörterbuchs vermieden wird. Diese zwei neuartigen Verbesserungen gegenüber momentan modernsten Methoden führen zu einer wesentlichen Verringerung sowohl des Rechenaufwands als auch des Speicherbedarfs. Daher ist unsere Methode skalierbar und kann auch für groSSe Datenbasen verwendet werden. Daneben entwickeln wir eine neue Methode zur Merkmalscodierung, die inkrementelle Codierung genannt wird. Zusammen können dieser neue Merkmalsextraktor und die inkrementelle Codierung bei der Klassifizierung von SAR-Bildern eine deutlich höhere Genauigkeit erreichen als modernste momentan bekannte Merkmalsextraktoren und deren Codierung. Zusätzlich wurden verschiedene Parameter im BoW-Verfahren untersucht; darauf aufbauend werden belastbare Schlussfolgerungen gezogen. Die BoW-Methode wurde auch auf Zeitreihen von SAR-Bildern erweitert, was zu einem neuen Bag-of-Spatial-Temporal-Words-Ansatz (BoSTW) führt - mit

einer höheren Leistungsfähigkeit als sie durch eine rein sequentielle Verkettung von extrahierten Texturmerkmalen erreicht wird.

Im letzten Teil der Arbeit wird ein Ansatz zum kaskadierten aktiven Lernen entwickelt, der auf einer Grob-zu-Fein-Strategie beim Mining von räumlicher und zeitlicher Information in SAR-Bildern beruht. Der Ansatz erlaubt in multi-temporalen SAR-Bildern eine schnelle Indexierung sowie die Entdeckung von bisher versteckten räumlichen und zeitlichen Mustern. Bei diesem Ansatz wird eine hierarchische Bilddarstellung verwendet und jeder Ebene wird eine eigene BildausschnittgröSSe zugeordnet. Um zuverlässige Klassifizierungsergebnisse zu erhalten und um den manuellen Aufwand beim Annotieren (Labeling) der Bildausschnitte zu reduzieren, wird auf jeder Ebene ein aktives Erlernen der Bildausschnitte mit Hilfe einer Support Vector Machine (SVM) durchgeführt. Dabei kommen abwechselnd zwei Komponenten zum Trainieren des Klassifizierers zum Einsatz: die Verwendung der bereits zugeordneten Bildausschnitte und die Beispielauswahl, die die aussagekräftigsten restlichen Bildausschnitte zur manuellen Annotation auswählt. Wenn der Prozess auf einer feineren Ebene der Kaskade fortgesetzt wird, werden alle bisher als negativ gekennzeichneten Bildausschnitte weggelassen und der Lernprozess auf der neuen Ebene beschränkt sich auf die positiv gekennzeichneten Bildausschnitte. Dadurch konnte der Rechenaufwand bei der Annotation von groSSen Datenmengen deutlich und ohne Genauigkeitsverlust reduziert werden. Bei dieser Methode konnten wir ein anderes Problem - die Weitergabe von Trainingsbeispielen zwischen Ebenen - durch Mehrfall-Lernen lösen. Dieses kaskadierte aktive Lernen wurde mit der Genauigkeit und der Zeitkomplexität eines Standard-SVM-Lernverfahrens auf der feinsten Ebene verglichen. Wir zeigen, dass kaskadiertes aktives Lernen nicht nur eine höhere Genauigkeit liefert, sondern auch die benötigte Rechenzeit deutlich reduziert. SchlieSSlich schlagen wir ein neues Visualisierungsverfahren für Zeitreihen von SAR-Bildern mit einer einfachen Farbanimation der Bildsequenz vor. Dabei werden jeweils drei aufeinanderfolgende SAR-Bilder zusammengefasst und in einer Sequenz von Rot-Grün-Blau-Farbbildern angezeigt. Die einfache Farbdarstellung kann Inhaltsänderungen deutlich hervorheben, ohne den Informationsgehalt zu verfälschen, was die visuelle Bildinterpretation stark vereinfacht. Dadurch können wir ohne weitere Bildverarbeitung viele zeitliche Muster einfach beobachten und Inhaltsveränderungen vollständig sehen.

Abstract

In this thesis, we focus on the development of new methods for spatial and temporal high resolution Synthetic Aperture Radar (SAR) image information mining. Starting from statistical models, we propose reliable models and robust methods for parameter estimation and evaluate statistical models on diverse classes of images. Based on the statistical models, information similarity measures are applied to SAR change detection both in the spatial and the wavelet domain. To evaluate their performance, a benchmark dataset is created by simulating changes, such as statistical changes in first, second, and higher order statistics, which resolves the problem of missing benchmark datasets for the comparison of various methods and allows a comprehensive evaluation of information similarity measures using both the synthetic dataset and real SAR data.

Based on the intrinsic characteristics of Very High Resolution (VHR) SAR images, two new feature extraction methods are developed. The first one represents a new approach for the structure description of high resolution SAR images, inspired by the well-known ratio edge detector. We apply brightness ratios in various directions of a local window in order to enhance the Bag-of-Words (BoW) feature extraction and to adapt a Weber local descriptor to SAR images. The second method is a simple yet efficient feature extraction method within the Bag-of-Words (BoW) framework. It has two main innovations. Firstly and most interestingly, this method does not need any local feature extraction; instead, it uses directly the pixel values from a local window as low level features. Secondly, in contrast to many unsupervised feature learning methods, a random dictionary is applied to feature space quantization. The advantage of a random dictionary is that it does not lead to a significant loss of classification accuracy yet the time-consuming process of dictionary learning is avoided. These two novel improvements over state-of-the-art methods significantly reduce both the computational effort and the memory requirements. Thus, our method is applicable and scalable to large databases. In parallel, we developed a new feature coding method, called incremental coding. Altogether, the new feature extractor and the incremental coding can achieve significantly better SAR image classification accuracies than state-of-the-art feature extractors and feature coding methods. In addition, several selectable parameters of the BoW method have been evaluated and reliable conclusions are given based on the evaluation. The BoW method has been extended to SAR Image Time Series (ITS) as well, resulting in a new Bag-of-Spatial-Temporal-Words (BoSTW) approach, which has shown a better performance than a simple sequential concatenation of extracted texture features.

In the last part of this thesis, a cascaded active learning approach relying on a coarse-to-fine strategy for spatial and temporal SAR image information mining is developed, which allows fast indexing and the discovery of hitherto hidden spatial and temporal patterns in multi-temporal SAR images. In this approach, a hierarchical image representation is adopted and each level is associated with a specific size of local image

patches. Then, Support Vector Machine (SVM) active learning is applied to the image patches at each level to obtain fast and reliable classification results and to reduce the manual effort to label the image patches. Within this concept, two components for classifier training work alternately: Using the already labeled image patches and a sample selection which selects the most informative remaining patches for manual labeling. When moving to a new finer level of the cascade, all the negative patches of the previous level are disregarded and the learning at the new level focuses only on the remaining positive patches. In this way, the computational burden in annotating large datasets could be remarkably reduced while preserving the classification accuracy. In addition, we solved the problem of training sample propagation between levels by multiple instance learning. We compared our cascade active learning with conventional SVM active learning operating only at the finest level in terms of classification accuracy and computational cost. It turns out that cascade active learning does not only achieve higher accuracy but also reduces remarkably the computation time.

Finally, we propose a new visualization method for SAR ITS using a simple color animation of the sequence. Successive triples of SAR images are represented as a sequence of red/green/blue coded color images. This simple color representation can significantly highlight the content variation of an image sequence without distorting the information content, which greatly facilitates the visual image interpretation. Without any processing, we can easily observe many temporal patterns and any content variation becomes completely visible.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

The last 10 years have witnessed a rapid development of new remote sensing SAR imaging sensors, including airborne (E-SAR, F-SAR, etc.) and spaceborne instruments (in particular, TerraSAR-X, TanDEM-X, and COSMO-SkyMed) with improved spatial resolution and short temporal re-visit intervals Soergel [2010]. As a consequence, both the image classes and the temporal variations are becoming quite diverse, leading to large image databases with rich information content. Short revisit times of a satellite allow us to acquire frequent successive observations of a given target area. Thus, SAR Image Time Series (ITS) can be built up, which represent a rich source of information about the dynamic evolution of the target area and have many potential applications in various fields. In contrast to single SAR images, an image time series contains a lot of information about the dynamic characteristics of a scene. An example of diverse temporal patterns occurring in images of agricultural fields is shown in Fig. 1.1. Here each of the nine rows contains a series of geographically overlapping sub-scenes acquired by the TerraSAR-X instrument when flying over the area of Sendai, Japan before and after the tsunami event in 2011. From left to right, one can observe the temporal evolution of detailed land cover patterns. The different types of temporal evolutions can be grouped into evolution classes. For instance, the nine examples depicted in Fig. 1.1 show nine different evolution classes in the field.

1.2 Goals

In this thesis, we try to develop new methods for temporal pattern extraction from SAR image time series within the context of information extraction and linked to the disciplines of machine learning, signal processing, estimation theory, information theory, and image retrieval. This is illustrated in Fig. 1.2 together with related application areas. The main goal of the thesis is to develop new methods for multi-temporal high resolution SAR image information mining based on the intrinsic characteristics of image data:

1. To develop new methods for multi-temporal SAR image analysis using information similarity measures based on statistical models and practical estimation methods for high resolution SAR images.

2. To develop novel feature extraction methods for the content representation of VHR SAR images based on their intrinsic characteristics.

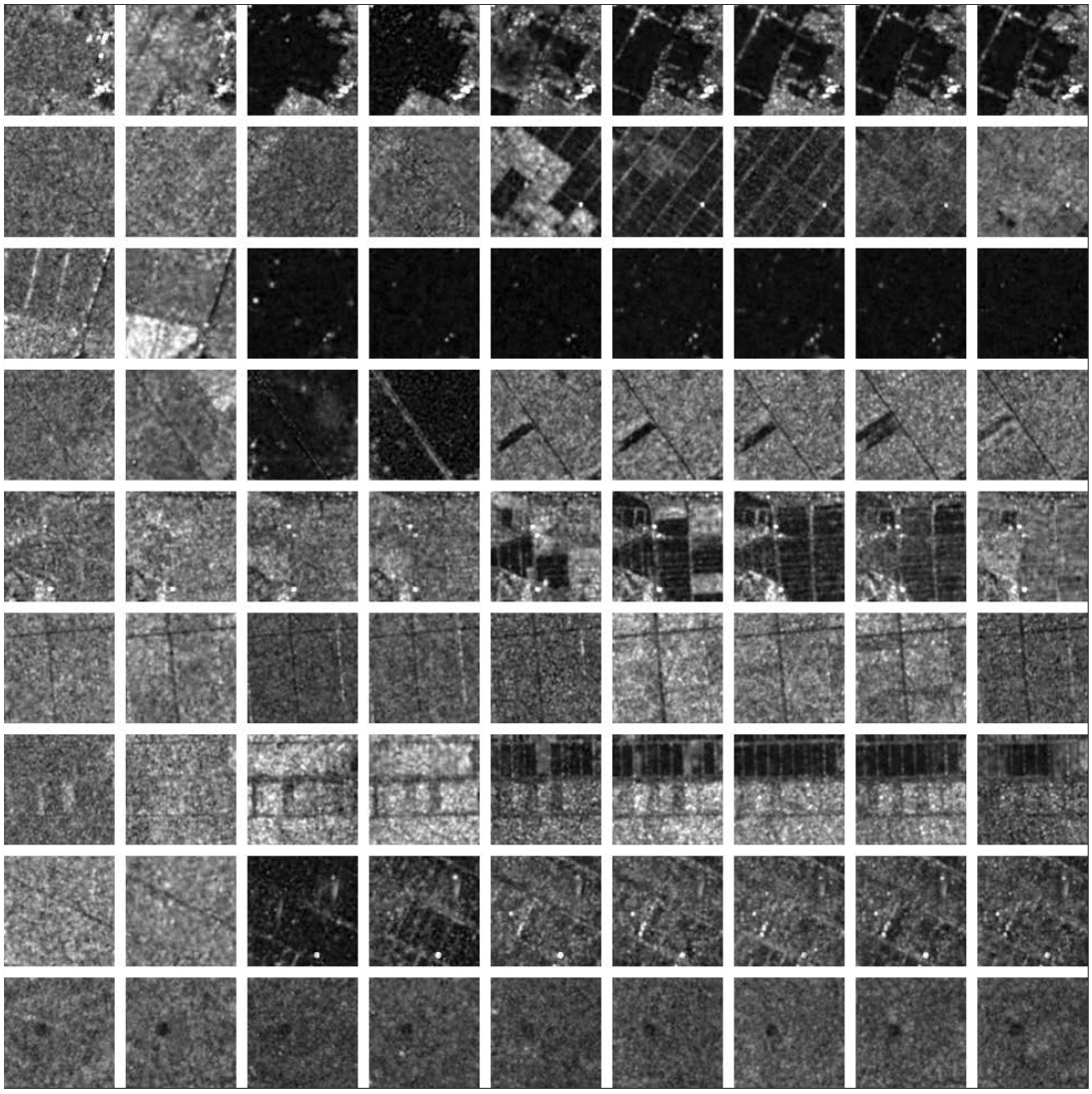


Figure 1.1: Diverse temporal patterns of agricultural fields: From left to right, each row illustrates the evolution of a particular temporal pattern.

3. To develop new learning approaches for fast indexing and the discovery of evolution patterns in SAR image time series.

1.3 Contributions of the Thesis

Based on the objectives listed above, the actual contributions of this thesis can be summarized as follows:

1. Starting from statistical models of high resolution SAR images, we propose reliable numerical methods for robust parameter estimation. The Method of Log Cumulants (MoLC) is applied as an estimation method that is better than conventional methods; this boils down to solving a set of constrained nonlinear equations. The Levenberg-Marquardt algorithm is

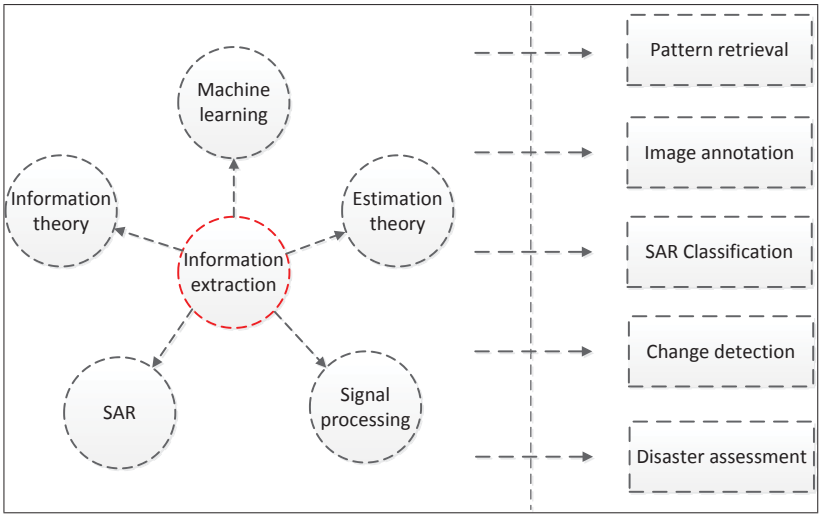


Figure 1.2: Surrounding context and applications of the thesis.

used to solve the constrained optimization problem and shows much reliability and robustness. We performed a comprehensive evaluation of statistical models using a quite diverse dataset consisting of 3582 TerraSAR-X images selected from 20 classes. From this evaluation, we found that for most heterogeneous image classes both the Generalized Gamma Rayleigh (GΓR) and the $\mathcal{G}^0$ distribution perform better than the $\mathcal{K}$ distribution. The Gamma distribution performs well only for images with fully developed texture. Empirical models such as the Generalized Gamma Distribution (GΓD) and the Log-normal distribution are also very good for some classes of images. The Generalized Gaussian Rayleigh (GGR) and the heavy-tailed Rayleigh distribution perform not as well as expected because of an involved numerical integral, which not only decreases the accuracy but also increases the computational burden.

We also studied information similarity measures for multi-temporal SAR change detection based on the statistical models. To evaluate their performance, a benchmark dataset was created by simulating changes, such as statistical changes in first, second, and higher order statistics. Thus, we obtained a benchmark dataset and we performed a comprehensive evaluation of information similarity measures using both the synthetic dataset and real SAR data. We found that the Kullback-Leibler method performs quite well in detecting intensity changes. In contrast, mutual, variational, and mixed information are better alternatives for changes in second order and higher order statistics. These information similarity measures can also be used as features in analyzing SAR ITS.

Due to the statistical similarity between the probabilistic density functions of SAR images and wavelet coefficients, we also applied the statistical models in the wavelet domain for SAR change detection. Given of their good mathematical properties, both the Generalized Gaussian Distribution (GGD) and GΓD were analyzed for wavelet coefficient modeling and we employed different estimation methods to estimate the model parameters. This study concludes that the resulting estimation accuracy strongly depends on the estimation method, although the statistical model is important as well. Specifically, the maximum likelihood estimation for GGD always has the lowest change detection accuracy. On the contrary, an estimation based on the shape equation for GGD improves significantly. In

most cases, an estimation of the GΓD by MoLC performs a little better than GGD in terms of accuracy and computational burden.

2. The intrinsic characteristics of VHR SAR images prompted the development of two new feature extraction methods. The first one is a new feature extraction method for the structure description of VHR SAR images, which is inspired by the ratio edge detector. Ratios in various directions within a local window are employed to enhance the Bag-of-Words (BoW) feature vector generation using only the local statistics. As ratios in the horizontal and vertical direction can be considered as an extension of image gradients, they are used to adapt the Weber Local Descriptor (WLD) to SAR images.

   The second method is a simple yet efficient feature extraction method within the Bag-of-Words (BoW) framework. It has two main innovations. Firstly and most interestingly, this method does not need any local feature extraction; instead, it uses directly the pixel values from a local window as low level features. Secondly, in contrast to many unsupervised feature learning methods, a random dictionary is applied to feature space quantization. The advantage of a random dictionary is that it does not lead to a significant loss of classification accuracy yet the time-consuming process of dictionary learning is avoided. Our method can achieve a better classification accuracy (around 91%) than other feature extractors (with an accuracy of less than 85%). Furthermore, we have developed a new feature encoding method, called incremental coding, which can improve the classification accuracy to 99%. In addition, we extended the BoW method to SAR ITS, giving a new Bag-of-Spatial-Temporal-Words method (BoSTW), which achieves a better performance than a concatenation of other texture features.

   The impact of other parameters in the BoW model have been evaluated comprehensively as well, such as patch size, patch sampling strategy, number of patches, universal or class-specific dictionaries, dictionary size, and feature coding methods. We drew clear conclusions about the BoW performance with respect to these parameters. We can show that a compact neighborhood size of 3×3 pixels is better than a large patch size. Regular dense sampling leads to a higher accuracy than random sampling with the same number of patches. Increasing the number of patches in random sampling can improve the accuracy, but is still inferior to regular sampling of small patches. A universal dictionary is better than the concatenation of class-specific dictionaries. However, the computing time of a universal dictionary is much longer than for a class-specific dictionary. As for the dictionary size, there is no obvious gain in accuracy by increasing the dictionary size as long as it is sufficiently large. Additionally, a large dictionary size would increase the computational burden. Through the comparison of feature encoding methods, we found that the accuracy gain of other methods is slight compared to vector quantization. However, our incremental feature coding method achieves significantly better results than the state-of-the-art methods, even in the case of quite less discriminative features.

3. Based on the observation that every category covers only a small part on an image, the learning method should discard all irrelevant patches as early as possible and focus on the training and learning of the relevant patches. In this way, the classification accuracy can be maintained, while the computational burden can be significantly decreased. Thus, we propose an enhanced cascade active learning approach relying on a coarse-to-fine strategy for spatial and temporal SAR image information mining, which allows the fast discovery of temporal patterns in SAR ITS. In this approach, a large fraction of irrelevant patches will be discarded when moving to a new level, which remarkably speeds up the learning.

We concentrate on a hierarchical image representation where each level is associated with a specific patch size. The patches are cut with smaller and smaller sizes in the hierarchy. To overcome the lack of training samples, we propose both SVM-based active learning and multiple instance learning (MIL) (see section 6.4). In SVM-based active learning, two important components are the modules for classifier training using the already labeled image patches and the sample selection which selects the most informative patches for manual labeling. These two components work alternatively, which can significantly reduce the human labeling effort and achieves a better performance for image indexing. In addition, we carried out a comparison of different sample selection strategies, which indicated that margin sampling performs best. We also solved the problem of training sample propagation between levels because the training samples are only available at the coarsest level. The training samples on the subsequent levels are automatically learned by MIL from the ones on the previous level, thereby further reducing the labeling effort. We compared this cascade active learning approach with a baseline SVM active learning performed only at the finest level through temporal pattern retrieval from a time series of TerraSAR-X images. We can demonstrate that the cascade active learning not only achieves better accuracy, but also reduces considerably the computing time.

Another important aspect is the visualization of SAR ITS. Normally, this includes dimension reduction. However, one of the biggest problems in applying these methods is that they distort the information content, such that a classifier or a user can no longer recognize the image content from the representation after dimension reduction. We solve this issue by a simple color animation of the image sequence. We concatenate triples of successive SAR images and represent them as color image and apply this representation to the entire image sequence. This simple color representation can significantly highlight any content variation without distorting the information content, thereby greatly facilitating the image interpretation. Without any processing, we can easily observe many temporal patterns and content variations become completely visible.

1.4 Outline of the Thesis

In **chapter 2**, we first present the intrinsic characteristics of VHR SAR images, followed by a description of the information content of VHR SAR image time series demonstrated by real examples. Thereby, the motivation of our research on SAR image time series and the need for the development of new methods, are demonstrated.

In **chapter 3**, we present the state-of-the-art of multi-temporal SAR image analysis. The chapter is composed of three main parts. The first part recalls statistical models and available statistical methods for SAR change detection, which are basic topics in multi-temporal SAR analysis. The second part focuses on SAR image feature extraction. We mainly review various texture feature extraction methods. The last part presents published techniques for satellite image information mining. In addition, dimension reduction methods for multi-dimensional data visualization are reviewed as well.

In **chapter 4**, we present statistical models and practical estimation methods for high resolution SAR images, followed by an evaluation using a diverse dataset. Then, information similarity measures are introduced. Based on the statistical models and estimation methods, information similarity measures are applied to unsupervised SAR change detection in the spatial and the transform domain. To assess the capabilities of information similarity measures for SAR change detection, a series of change simulations is carried out, resulting in a set of synthetic data. Due

to the statistical similarity between the probability density functions of SAR images and wavelet coefficients, these statistical models are also applied and evaluated for SAR change detection in the wavelet domain.

In **chapter 5**, we propose two novel feature extraction methods for high resolution SAR images. The first one is a new feature extraction method based on the ratio edge detector. Ratios in various directions are applied to enhance the BoW feature vector based on local pixel statistics and to adapt the WLD algorithm to SAR images. The second method is a simple yet efficient feature extraction method within the Bag-of-Words (BoW) framework, which has two main innovations. Furthermore, we develop a new feature coding method called incremental coding. In addition, the parameters involved in the BoW method are evaluated rigorously, like the sampling strategy, patch size, and dictionary size. In the last part of this chapter, the BoW method is extended to SAR image time series; this extension performs better than a concatenation of all conventional features.

In **chapter 6**, a cascade active learning approach relying on a coarse-to-fine strategy for spatial and temporal SAR image information mining is described, which allows fast indexing and the discovery of hidden spatial and temporal pattern in multi-temporal SAR images. Three different learning methods are presented. An evaluation and comparison of cascade active learning using a time series of real SAR images is performed. In addition, a comparison of sample selection strategies in active learning is given. A simple yet efficient visualization method for multi-temporal SAR images is proposed as well.

Finally, **chapter 7** draws some conclusions summarizing the main results of the thesis and suggests some perspectives for future work.

Chapter 2

Information Content of High Resolution SAR Image Time Series

In this chapter, we first present the intrinsic characteristics of VHR SAR images, followed by a description of the information content of high resolution SAR ITS based on examples of TerraSAR-X images. These examples prompted us to develop new information mining techniques for high resolution SAR image time series.

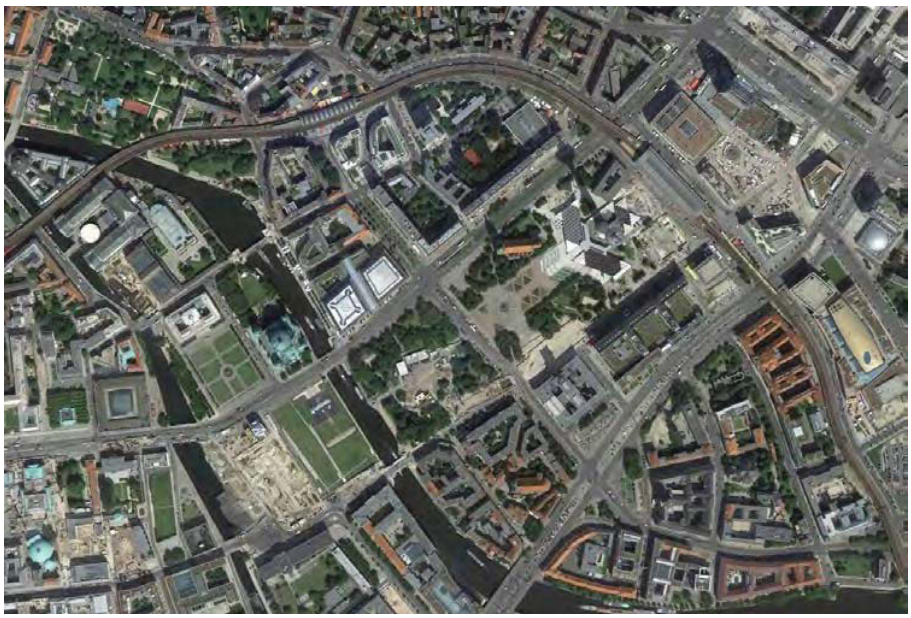
2.1 Characteristics of VHR SAR Images

A sub-scene of a typical TerraSAR-X image (with a pixel spacing of about 3 m) covering the inner city area of Berlin, Germany is shown in Fig. 2.1. To support its visual interpretation, a high resolution optical image provided by Google Earth is also presented. The VHR SAR image is characterized by groups of bright and dark pixels. A lot of information about the image content is plainly visible and a large number of objects can be identified and recognized. This opens the path for a number of potential Earth Observation (EO) applications. The typical characteristics of VHR SAR images are the local context consisting of complex structure arrangements, resulting in a high number of scene classes. However, only groups of pixels have a semantic meaning. It is very hard to interpret the image data, not only due to the complex coherent imaging mechanics of SAR sensors, but also due to the complex local context. In contrast to medium and low resolution SAR images, the stationarity assumption of the image content in terms of texture is no longer valid for urban areas. Therefore, texture features, which can only discriminate classes with fully developed speckle, like forest and agriculture, are not sufficient for a VHR SAR image content characterization. Similarly, conventional SAR image analysis techniques, like image segmentation and pixel level classification, are likely to fail since high level semantic descriptions of the local context have to be taken into account for discrimination. We have to consider the complex structure arrangements in their local context; otherwise, a scene description cannot be efficiently achieved.

2.2 Information Content of VHR SAR ITS

The increasing number of spaceborne SAR imaging sensors and their improved resolution has led to large volumes of SAR data being available in various Earth observation centers. Moreover, a target area can be observed repeatedly within a short period of time, thus enabling the creation of SAR ITS. In contrast to single images, SAR ITS contain highly detailed spatial and temporal

(a)



(b)

Figure 2.1: Satellite images of Berlin, Germany. (a) A sub-scene of a very high resolution TerraSAR-X image. (b) The same target area seen by Google Earth.

information on the dynamic characteristics of the target area. They are therefore highly complex data containing numerous and various spatio-temporal information, which have a great potential for EO applications. For instance, the growth status of crops can be monitored by SAR ITS as described in the literature Lopez-Sanchez *et al.* [2011]. Typically, we can observe seasonal

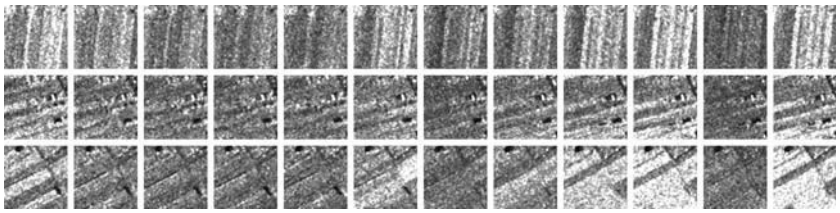
vegetation changes or the evolution of crop rotations in agriculture. An example of crops having been observed in summer is shown in Fig. 2.2(a). The image patches are selected from a sequence of 12 TerraSAR-X images covering the Vâlcea County in Romania with a revisit time of 11 days. The standard deviations of all pixels of each patch in the sequence are plotted in Fig. 2.3(a). Apparently, there are temporal patterns in crop growth. Similarly, another example sequence of grassland patches (also acquired in summer) is shown in Fig. 2.2(c); the corresponding standard deviations of all patches versus time are plotted in Fig. 2.3(c). For these two categories, i.e., crops and grassland, an evolution pattern due to crop growth can be clearly observed. The growing periods of these two categories are quite short compared with forest, with an example shown in Fig. 2.2(b) and the corresponding standard deviations depicted in Fig. 2.3(b). Compared with crops and grassland, forest is more stable, thereby no abrupt changes can be seen in the standard deviations.

On the contrary, some other categories from the urban areas are more stable. Three sequences of residential areas with buildings are shown in Fig. 2.2(d); the corresponding standard deviations are shown in Fig. 2.3(d). It can be clearly seen that there are no obvious fluctuations in the evolution patterns. This is expected because residential areas with buildings do not change much in a short period of time. Thus, the corresponding evolutionary patterns are contrast with crops, grassland, and forest. However, if the images contain some vegetation, we would see some fluctuation reflecting its growth patterns.

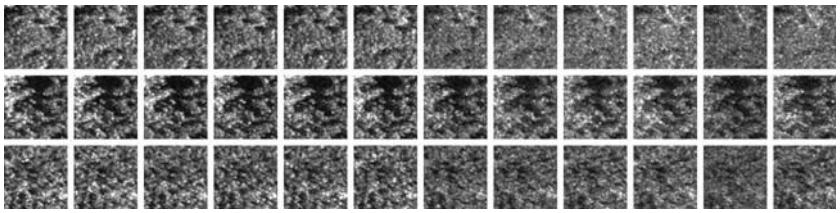
Other frequent patterns in EO images are due to abrupt changes, which usually occur after various disasters, like earthquakes, floodings, or tsunamis. An example of building destruction by the Japanese earthquake and tsunami in 2011 is shown in Fig. 2.2(h). Eight observations were acquired, one before the disaster, and seven after it. Post the disaster, the buildings are destroyed. Before and after the disaster, there is an obvious abrupt change, as can be seen from the standard deviation plot in Fig. 2.3(h). Here, an extension of bi-temporal change detection shall detect the abrupt changes in the entire SAR ITS. However, compared to categories like grassland or agriculture, it is not trivial to find a method to describe the patterns of destroyed buildings.

The earthquake in Japan and the subsequent tsunami resulted in devastating floodings. A large part of agricultural fields close to Sendai were flooded to a different degrees and some rivers and lakes overflowed due to the increasing water level. Two examples of flooded fields are presented in Fig. 2.2(e) and (f). In the first series, the target area is close to the sea shore, thereby being flooded all the time. The second series depicts a patch further away from the sea shore than the first one, thus not being flooded during the first four observations. After that, an abrupt change is observed. From these two examples, we can see that SAR ITS may contain different evolution patterns. An example of an overtopped river and lake is shown in Fig. 2.2(g), together with the evolution of the standard deviations in Fig. 2.3(g). From the standard deviation plot, we see that the flooding creates similar evolution patterns.

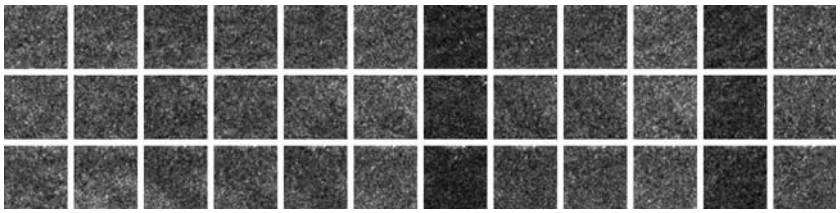
As can be seen from the eight SAR ITS examples presented above, a SAR ITS contains much more information than a single image and reflects the dynamic evolution of the observed scene. When we consider temporal relationships, we see many more patterns than in single images. Each evolution class has a particular pattern representing a particular semantics due to a certain event. Thus, SAR ITS shows promise for various monitoring applications in the context of Earth Observation. However, in contrast to single images, there are some great challenges when we develop new methods for the discovery of evolution patterns from SAR ITS. Available methods in the literature focus on change detection and pixel-wise classification. As long as there are reference data available, this problem can be easily formulated as a classification problem. However, reality is never so simple. First of all, reference data for VHR SAR ITS are not easy



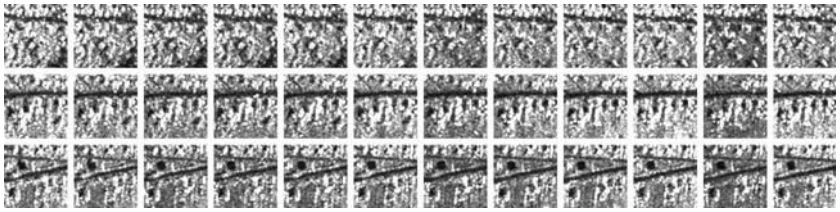
(a) agriculture evolution



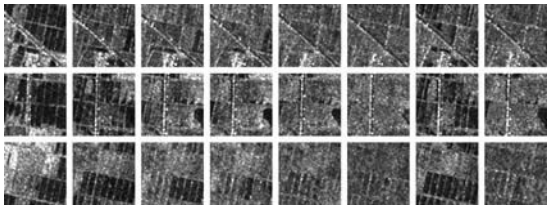
(b) forest evolution



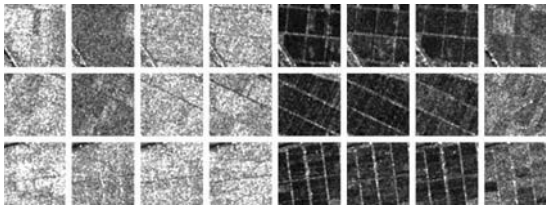
(c) grassland evolution



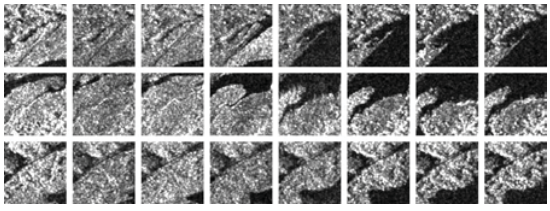
(d) building evolution



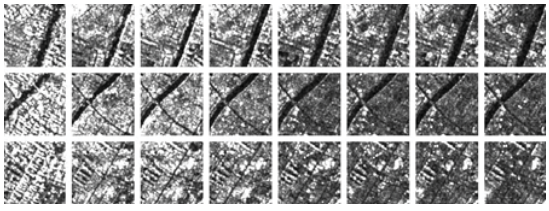
(e) flooding evolution



(f) evolution of flood type 2



(g) evolution of overtopped lake



(h) evolution of building destruction

Figure 2.2: Examples of SAR ITS (three examples for each category; time runs from left to right).

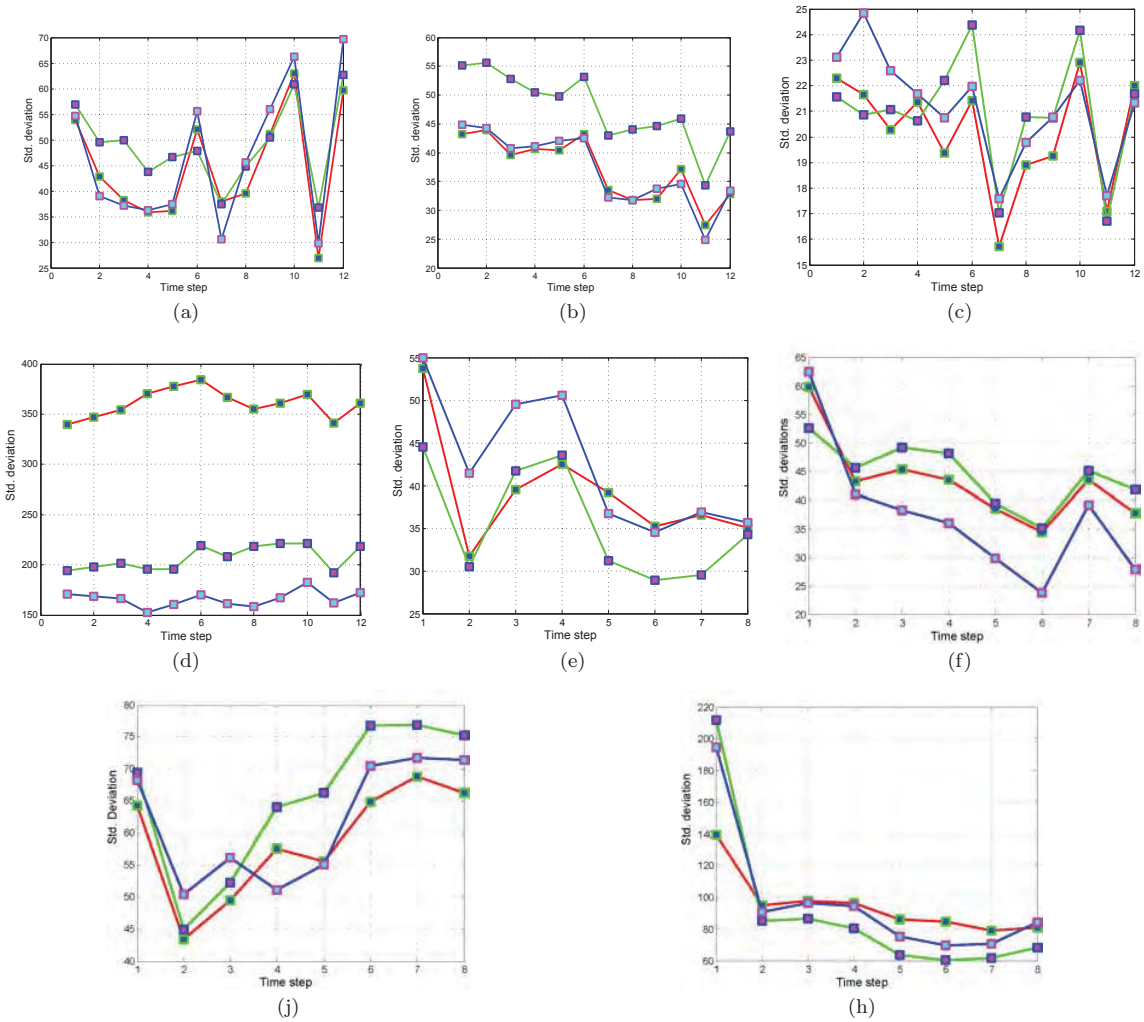


Figure 2.3: Evolution of various temporal patterns in SAR ITS: Standard deviations of categories vs. time. (a) Std. deviation evolution of agriculture; (b) Std. deviation evolution of forest; (c) Std. deviation evolution of grassland; (d) Std. deviation evolution of residence; (e) Std. deviation evolution of flooded field; (f) Std. deviation evolution of flood type 2; (j) Std. deviation evolution of lake; (h) Std. deviation evolution of destroyed residence.

to obtain. In practice, we have to resort to manual interpretation, which becomes unfeasible for large scale applications. Furthermore, the results reached by a pixel-wise classification of high resolution SAR data would not be meaningful, even when reference data are available, due to the nature of SAR that differs from optical imaging. For instance, if a classification is performed on SAR images of an urban area, all bright and dark pixels would be separated as two different categories. However, this is not meaningful because some bright pixels in SAR images depict strong single reflection or double bounce scatterers, but they do not represent a valid physical category. In a similar sense, any segmentation would fail as well. Therefore, for SAR image interpretation, we propose to use image analysis methods based on local patches that take the local image context into account.

One practical problem in applications is that we usually do not know what and how many patterns exist in the database. We have to resort to additional reference data, which means that our dataset could not be used to generate the missing reference data with our current techniques. As a consequence, data mining techniques for SAR ITS relying on machine learning, like content-based retrieval methods, are highly demanding, and this is the main focus of this thesis.

Chapter 3

Multi-Temporal and ITS Analysis: State-of-the-Art

In this chapter, we briefly review related work on multi-temporal SAR image analysis methods, including SAR statistical models, SAR change detection, SAR feature extraction, SAR ITS analysis, and related image information mining techniques. The basic methods for analyzing SAR images are statistical models, which are our starting point for a literature review in section 3.1. SAR change detection methods relying on statistical models are recalled in section 3.2, as well as other change detection methods. As a fundamental part in the development of learning algorithms, SAR image feature extraction is reviewed in section 3.3, focusing on texture features. Satellite image time series analysis methods are reviewed in section 3.4, while image information mining methods are reviewed in section 3.5. Since the visualization of multi-temporal SAR images is an important topic, dimension reduction methods are reviewed in section 3.6. Finally, a conclusion on the state-of-the-art is drawn and our solution is proposed.

3.1 SAR Statistical Models

SAR statistical modeling is a basic problem in SAR image analysis; it describes an image and reveals its characteristics through statistical methods. It plays a fundamental role for many other processing approaches, like despeckling Achim *et al.* [2006]; Daniela Espinoza-Molina [2012], edge detection Touzi *et al.* [1988], change detection Ban & Yousif [2012]; Moser & Serpico [2006]; Wang *et al.* [2012], segmentation Bombrun *et al.* [2011]; Feng *et al.* [2013]; Galland *et al.* [2009]; Marques *et al.* [2012], classification Krylov *et al.* [2011]; Tison *et al.* [2004]; Voisin *et al.* [2013], and target detection Gao & Shi [2012]; Gao *et al.* [2009]. An extensive survey of statistical SAR image models was published by Gao [2010]. In this section, we give a brief review of existing statistical models and the latest developments. Theoretically, statistical models fall into three categories: non-parametric, semi-parametric, and parametric models. Non-parametric models do not assume any predefined functional form of the unknown probability density function; thereby, the models are estimated in a data-driven way. The drawback of non-parametric models is the computational burden; their advantage is the attainment of high accuracy (if sufficient samples are available). The semi-parametric models, usually referring to various mixture models, impose certain relaxed constraints on their functional form. In this section, we focus on parametric models based on the physical principles of SAR imaging. In the literature, most models originate from the speckle model of Arsenault & April [1976] or the product model of Ward [1981]. The speckle model is based on the assumptions that the Radar Cross Section (RCS) is constant and

the speckle produces a fully developed stationary texture. On the basis of these assumptions and the central limit theorem, both the real and imagery components of a complex-valued SAR image will follow a Gaussian distribution. Accordingly, single look amplitude images will have a Rayleigh distribution; their intensity follows a negative exponential distribution Oliver & Quegan [1998]. However, when the resolution of a SAR image increases, one can often observe that these assumptions do not hold any longer. Consequently, several other models have been proposed to model the real and imagery components of a SAR image, like the alpha-stable distribution of Kuruoglu & Zerubia [2004], the Generalized Gaussian Distribution (GGD) of Moser *et al.* [2006], or the Generalized Gamma Distribution (GΓD) of Li *et al.* [2010]. Corresponding models for the amplitude and intensity can be derived by the simple mathematical relation between the two components.

The product model is a generalization of the speckle model by considering the radar echo as the product of speckle and backscatter and also allows variable RCS values. Thus, seeking for a statistical model is equivalent to finding individual models for speckle and backscatter respectively. Equivalently, if the speckle intensity is assumed to follow a Gamma distribution and the backscatter is a constant, we would arrive at the same amplitude and intensity distribution as the ones derived from the speckle model. In most cases, the number of scatterers in each elementary cell for a homogeneous region is assumed to be large enough and approximately constant. However, if the resolution increases, the number of backscatters in a resolution cell decreases and the RCS will no longer be constant. Assuming that the random number of scatterers in each elementary cell is distributed according to a Poisson distribution, one can consider that the mean of this distribution, or the expected number of scatterers, is itself a random variable Jakeman [1980]. If a Gamma distribution is assumed for the mean, the SAR images would follow a $\mathcal{K}$ distribution Oliver [1993], which is a breakthrough in modeling heterogeneous regions and has been widely used in various applications. Inspired by the deviation of the $\mathcal{K}$ distribution, Delignon & Pieczynski [2002] proposed another three alternative models. If the mean is assumed to follow an inverse Gamma, Beta of the first kind, or Beta of the second kind relationship, the SAR intensity of a heterogeneous region would follow a B, U, or W distribution. Furthermore, a $\mathcal{G}$ model was proposed by Frery *et al.* [1997] for extremely heterogeneous regions by assuming that the mean follows a generalized inverse Gaussian distribution, which includes the $\mathcal{K}$ distribution as a special case. As a special case of the $\mathcal{G}$ model, the $\mathcal{G}^0$ model is highly suited to model extremely heterogeneous surfaces, such as urban areas. Coincidentally, the empirical Fisher distribution model F is equivalent to the $\mathcal{G}^0$ distribution.

In addition to these two models, there are also a number of empirical models for SAR images, like the Weibull distribution Oliver [1993], the Log-normal distribution Oliver & Quegan [1998], the Nakagami Rice distribution Oliver & Quegan [1998], and the GΓD of Li *et al.* [2007, 2010]. Among these empirical models, GΓD shows promising ability in modeling heterogeneous regions with a closed-form approximation estimator. When referring to statistical models, we have to find an efficient estimation method for parameter estimation. With the exception of conventional estimation methods, such as Maximum Likelihood Estimation (MLE) and the Method of Moments (MoM), the Method of Log-Cumulants (MoLC) of Nicolas [2002] has become the state-of-the-art for SAR image probability density estimation.

3.2 SAR Change Detection

As a basic topic in multi-temporal SAR image analysis, change detection has been continuously studied for many years. Consequently, many methods have been proposed to address this problem. In this section, we review several change detection methods, both for SAR images and

optical images, because some methods have common elements for both kinds of images. From the point of view of the data sources, they can be classified into bi-temporal change detection methods Bovolo & Bruzzone [2005] and image time series change detection methods Salmon *et al.* [2011] Chen *et al.* [2011]. Most bi-temporal change detection methods fall into two categories: supervised Bruzzone & Prieto [2002] and unsupervised change detection Bruzzone & Prieto [2000]. Both kinds of methods were extensively evaluated for detecting flooded areas in the 2009-2010 data fusion contest described by Longbotham *et al.* [2012]. In supervised change detection, reliable training samples based on prior knowledge about the research scenario are selected and used to train a classifier, which after training will be used to classify each pixel as a changed or unchanged pixel. In contrast, in unsupervised change detection, the first step is to compare the two images or some extracted features by some similarity metrics resulting in a change map, and then to threshold or label the change map in order to derive a binary change map consisting of two classes associated with changed and unchanged pixels. Therefore, in practice, unsupervised approaches are preferable to supervised approaches since training samples are not always available.

Supervised change detection can be considered as a binary classification including post-classification comparison Singh [1989], Hall *et al.* [1991], direct multi-data classification Singh [1989], Jeon & Landgrebe [1992] and compound classification Bruzzone & Prieto [2001]; Bruzzone & Serpico [1997]; Solberg *et al.* [1996]. A post-classification comparison performs change detection by analyzing the classification maps derived by an independent classification of the two images, while a direct multi-data classification generates a change map by classifying the concatenated features of the two images. Compound classification performs the change detection by maximizing the posterior joint probabilities of classes, which can be derived by a specific estimation method. Recently, Support Vector Machines (SVMs) are widely applied to supervised change detection. In Huo *et al.* [2010], an inductive SVM was used for classifying features characterizing changes at an object level; then the classification was refined by an iterative transductive SVM. As an alternative, the ν-SVM using a stochastic kernel published by Mercier *et al.* [2006] was applied to change detection based on the similarity measures of the local statistics. Another approach to unsupervised change detection was proposed in Bovolo *et al.* [2008] using a selective Bayesian thresholding technique to obtain a training set for initializing a binary semi-supervised SVM classifier. Finally, in Bovolo *et al.* [2010], change detection was formulated as a minimum enclosing ball problem which was approached by a support vector domain description classifier. One advantage of these methods is that they can capture different changes denoted by a transition map. However, in practice, training samples are not always available and it is expensive both in terms of time and resources to obtain reliable training data.

For most methods, image comparison is an important step. One widely used technique for SAR image comparison is the ratio operator Bazi *et al.* [2005]; Çelik [2010]; Rignot & van Zyl [1993] which is especially designed for SAR change detection. Evidentially, the ratio operator is more efficient for SAR change detection, than the image subtraction that is widely used for change detection in optical images. Many techniques have been proposed to analyze the change map derived by the ratio operator in order to obtain a final binary change map associated with changed and unchanged classes. The most intuitive method is to threshold the change map. Bruzzone & Prieto [2000] proposed two automated techniques for the analysis of difference images based on Bayesian theory. The first method aims at automatically selecting a threshold in order to minimize the overall change detection error, and is based on the assumption that the change map consists of two Gaussian components associated with changed and unchanged classes. This idea was extended in Bazi *et al.* [2007] to a generalized Gaussian mixture model estimated by an Expectation-Maximization (EM) algorithm. The second method performs better

as it considers the thresholding problem as an image classification task approached by a Markov random field model, which takes contextual information into account. An extension of the second method was presented by Bruzzone & Prieto [2002] who introduced an adaptive semi-parametric approach for conditional density estimation, which was formulated as a two component Gaussian mixture model. The same framework was used by Çelik [2010] in the wavelet domain instead of the spatial domain. The differences between the methods lie in their statistical models, such as the function form and the parameter estimation.

In recent years, promising methods based on information measures have been developed for multi-temporal change detection due to their efficiency and simplicity. Most of these methods are based on the idea that image pairs acquired over the same area at two different times are two measurements of the same information source. Therefore, several information similarity measures have been proposed to assess the similarity of different measurements of the same source. The prominent work of Inglada & Mercier [2007] proposed a new method for multi-temporal SAR change detection based on the evolution of the local statistics of the two images. The local statistics are estimated by a one-dimensional Edgeworth series expansion, which approximates the probability density functions in the neighborhood of each pixel. The degree of evolution of the local statistics is measured using the Kullback-Leibler divergence. In Bovolo & Bruzzone [2008], this method was extended to object-based change detection by computing the Kullback-Leibler divergence of the two corresponding objects extracted by image segmentation. Kullback-Leibler divergence was used in Atto *et al.* [2013] to construct a multi-date divergence matrix for the detection of changes in SAR image time series.

In Alberga [2009], several information theoretical similarity measures including distance to independence, mutual information, cluster reward algorithm, the Woods criterion and correlation ratio, were compared for change detection. Here, the mutual information based similarity measure proved to be rather efficient. Mutual information has also been widely used for image analysis, especially for image registration Chen *et al.* [2003]; Inglada & Giros [2004]; Kern & Pattichis [2007]; Suri & Reinartz [2010], content-based image retrieval Faur *et al.* [2006]; Liu *et al.* [2008]; Tourassi *et al.* [2007], and change detection Alberga [2009]; Chatelain *et al.* [2007]; Gueguen *et al.* [2011a]; Mercier *et al.* [2006]; S. Le HÃľgarat-Mascle [2004]. In S. Le HÃľgarat-Mascle [2004], mutual information, as a change index, was fused with normalized difference values and texture features for forest fire detection.

Taking advantage of mutual information, a pixel-based approach comparing localized mutual information was proposed in Winter *et al.* [1997]. Intuitively, when two pixels share a lot of information, it is reasonable to assume no change at their location. Based on this idea, another information measure for change detection derived from mutual information was introduced in Gueguen & Datcu [2009]; Gueguen *et al.* [2011b]: namely the mixed information which unifies mutual information and variational information. This measure introduces a new parameter to control the balance between common and different information. Furthermore, stochastic kernels including both Kullback-Leibler divergence and mutual information were used in Mercier *et al.* [2006] as a feature vector in the context of a SVM for SAR change detection. Based on the estimation of a bivariate Gamma distribution, mutual information was applied to SAR change detection in Chatelain *et al.* [2007]. A region-based local mutual information change indicator was proposed by Gueguen *et al.* [2011a] to perform a change analysis of urbanization processes from multi-temporal panchromatic SPOT 5 images. Through a 2-scale implementation, mutual information can be split into two terms to be linked to a change detection part and a registration part Mercier & Inglada [2008].

Apart from the spatial domain, SAR change detection can also be performed in a transformed domain, such as the wavelet domain. The method described in Bovolo & Bruzzone [2005] exploits

a wavelet-based multi-scale decomposition of a log-ratio image aiming at the representation of the change signal using different scales. First, this method decomposes the log-ratio image using a two-dimensional discrete stationary wavelet transform and selects important scales according to an adaptive analysis of its local statistics. Then, a change map is derived by a combination of scale-driven changes according to three different strategies. In Çelik [2009], k-means clustering was applied to classify the features extracted by an undecimated wavelet transform of the difference image into two classes corresponding to change and no-change classes. In Çelik [2011], Bayesian inference was applied to the difference image for change detection, which is an extension of the method published in Bruzzone & Prieto [2000]. Similar methods performed in the wavelet domain were proposed in Çelik [2010] Çelik & Ma [2010]. Here, a wavelet transform is applied to decompose the log-ratio image into multiple levels. Probabilistic Bayesian inference with expectation maximization parameter estimation is applied to each scale in order to derive a change map, which is similar to the first method proposed in Bruzzone & Prieto [2000] and the method in Bazi *et al.* [2007]. The final binary change map is formed by merging intra-scale and inter-scale information. As an alternative to these statistical methods, an active contour segmentation of the wavelet sub-bands for change detection was proposed in Çelik & Ma [2011].

3.3 SAR Image Feature Extraction

In this section, we briefly review methods for SAR feature extraction with a focus on texture. Texture description has played an important role in SAR image understanding for years. Consequently, a large variety of texture features have been developed. A texture is a class of images consisting of repetitive patterns, called "Textons" by Julesz [1981], which can be seen frequently in SAR images, as shown in Fig. 3.1. Haralick *et al.* [1973] have proposed to use statistics computed from the gray level co-occurrence matrix (GLCM) for texture description. The co-occurrence matrix is the joint distribution of pixel pairs with a certain offset in a direction and represents the second order statistics of an image. The statistics, like contrast, entropy, homogeneity, etc., can be computed based on the co-occurrence matrix and used as a feature vector. Later, model-based texture description methods, like Markov Random Fields, were developed, which consider an image as a realization of a stochastic random process Cross & Jain [1983]. The parameters governing the stochastic models are estimated and used as texture features. A number of Gaussian Markov Random Fields (GMRFs) have been developed over the years for SAR image interpretation Clausi & Yue [2004]; Daniela Espinoza-Molina [2012]; Gleich & Datcu [2007]; Walessa & Datcu [2000]. Evidence from cognitive psychology suggested that Gabor filter banks are able to capture the properties of human image understanding Marcelja [1980] . Therefore, the statistics of Gabor filter responses with different orientations and scales have received a lot of interest for texture characterization following Manjunath & Ma [1996], which is still a competitive technique today. Following the same idea, a lot of methods in transformed domains have been developed especially for wavelet Akbarizadeh [2012]; Fukuda & Hirosawa [1999]; Gleich & Datcu [2009]; Karvonen & Simila [2002].

Although texture can be easily recognized and is defined as a repetitive pattern of primitives, there are still no formal mathematical models available. Recently, the analysis granularity has been moved from pixels to local patches, which is considered as being close to the concept of "Textons" Leung & Malik [2001]; Varma & Zisserman [2005]; Zhu *et al.* [2005]. Inspired by previous research that texture can be efficiently characterized by the statistics of filter responses, some filter banks, like the Leung-Malik filter bank and the maximum response filter set were applied to texture description. It has two steps: a training phase and a test phase. In the training phase, a set of training images $I_i, i = 1.., n$ from each class are convolved with each

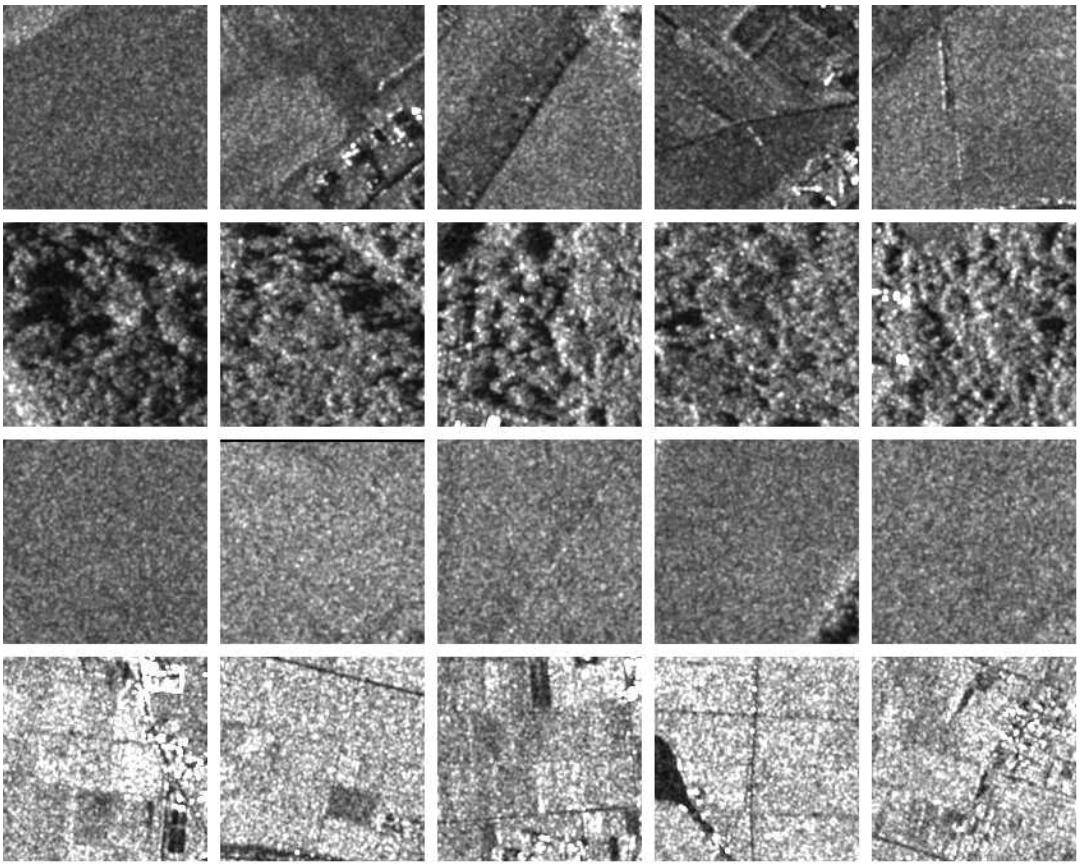


Figure 3.1: Examples of homogeneous textures in SAR images.

filter $F_i, j = 1, ..., L$ of the filter bank. Thus, L filter responses with the same size as the given image are produced. A subset of the filter responses from each class are clustered to seek for K clusters. The cluster centers that form a dictionary D_i are considered as the texture model of this texture class. All the dictionaries are then concatenated into a universal dictionary. In the test phase, the test images are also convolved with the filter banks. Using the universal dictionary and the filter responses, the images will be labeled by a mapping of the filter responses to the elements in the dictionary based on the nearest neighbor search. Finally, a texton histogram of the labels can be generated as a feature vector. The entire procedure is shown in Fig. 3.2. The most important problem is to select a good filter bank. For example, 48 filters were proposed by Leung & Malik [2001], which are first and second order derivatives of Gaussian filters with 6 orientations and 4 scales together with 8 Laplacian of Gaussian filters, and 4 Gaussian filters. 38 filter banks were developed by Varma & Zisserman [2005], which include a Gaussian filter, a Laplacian of Gaussian filter, an edge filter with 6 orientations and 3 scales, and a bar filter with 6 orientations and 3 scales.

Since the publication of the Scale Invariant Feature Transform (SIFT) proposed by Lowe [2004], local feature descriptors have received much attention and a large amount of research effort was directed to local feature extraction. Inspired by both the texton image representation and the power of local feature descriptors, the Bag-of-Words (BoW) method was proposed by Sivic & Zisserman [2003] for video search. Since then, under this framework, a large variety of methods

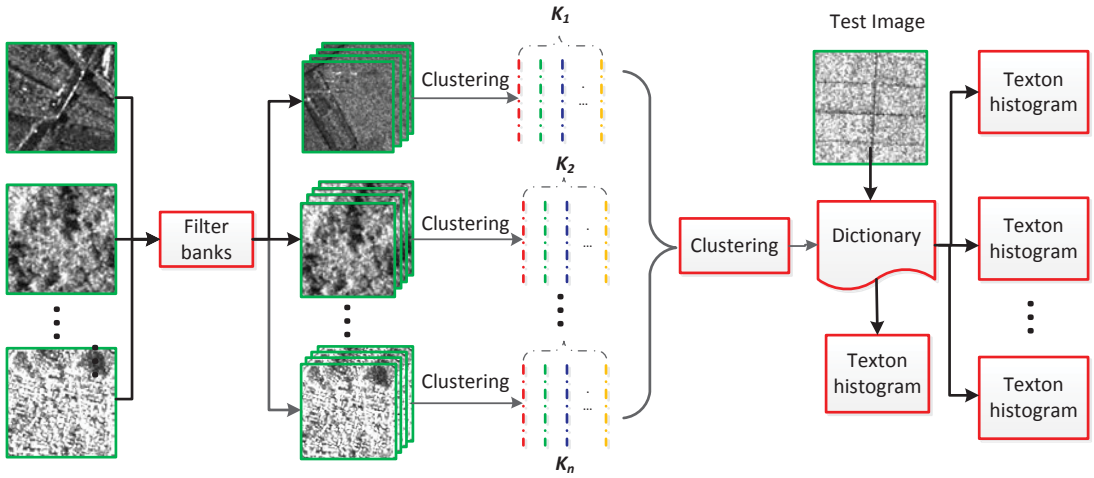


Figure 3.2: Texton feature extraction.

have been proposed for solving various problems, like image classification, image retrieval, and object recognition. It has been recently introduced to the remote sensing community for image annotation by Lienou *et al.* [2010], object classification Xu *et al.* [2010], target detection Sun *et al.* [2012] and land use classification Yang & Newsam [2010]. The BoW model has proved its ability in efficient image classification. The framework of the BoW model is shown in Fig. 3.3, which comprises five main components: feature detection, local feature extraction, codebook learning, feature coding, and feature pooling. All these five steps have received a large amount of research effort. The SIFT detector is one of the most widely used methods for feature detection, and it detects sparsely distributed keypoints for local feature extraction. Partially inspired by SIFT, the Speeded Up Robust Feature (SURF) was proposed by Bay *et al.* [2008], which can be computed faster and is more robust to image transformations. Nevertheless, there are some works Maree *et al.* [2005]; Nowak *et al.* [2006] showing that dense sampling or even random sampling can achieve better performance than the SIFT detector. Nowak *et al.* [2006] have shown that random dense sampling performs better than the SIFT detector as long as the number of patches is sufficient. Maree *et al.* [2005] demonstrated that randomly extracted sub-windows perform better in image classification. Lazic & Aarabi [2007] compared various sampling strategies and concluded that a simple variance-based point selection method can be more effective than using a regular grid, random points, or the SIFT detector. On the other hand, many local feature descriptors have been proposed, like SPIN image and Rotation Invariant Feature Transform (RIFT) proposed by Lazebnik *et al.* [2005], Census transform histogram (CENTRIST) Wu & Rehg [2011], and various Local Binary Patterns (LBP) Ojala *et al.* [2002]; Zhao *et al.* [2012]. The codebook (or dictionary) in the BoW method is usually generated by clustering. One problem of the BoW method is that it does not use spatial information. Thus, the Spatial Pyramid Match (SPM) has been developed by Lazebnik *et al.* [2006] to incorporate spatial information. In this method, the images are split into multiple regions in different scales and the local word histogram of each region is computed. All the word histograms are concatenated to form a vector representation of an image. As SPM is designed to incorporate spatial information for natural scene classification, it may not be fully applicable to SAR image classification, because there is no fixed spatial layout information available for SAR images, such as wheels are always below a car and a head is always on the top of a person. Recently, sparse coding Yang *et al.* [2009] instead of vector quantization has been applied to dictionary learning, which shows superior performance.

Unfortunately, sparse coding is computationally expensive. As for the word assignment, there are two kinds of methods: hard assignment and soft assignment. Hard assignment is to assign a feature vector to its nearest element in the dictionary. However, it has been shown by van Gemert *et al.* [2010] that soft assignment (section chapter 5.3.3) can improve the classification accuracy. All these research activities are still on going. We focus on these issues in chapter 5 and provide some clear answers in the context of SAR image classification.

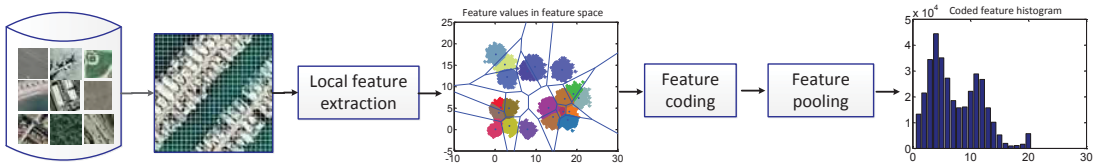


Figure 3.3: The framework of the Bag-of-Words model consists of five steps: patch sampling, local feature extraction, dictionary learning, feature coding, and feature pooling.

3.4 Satellite Image Time Series Analysis

In the context of Earth observation, time series data analysis is probably best known by ozone analysis Michelle L. Bell [2005]; Omidvari *et al.* [2008]. Advanced satellite image time series analysis is a very young branch in Earth observation land cover applications, therefore only few works are more than 10 years old. In this section, we review related works on this topic. Earlier works on this topic mainly focus on analyzing the dynamical patterns of vegetation. Hidden Markov models were applied by Viovy & Saint [1994] to extract various vegetation dynamics using satellite remote sensing. Multi-temporal SAR images of the European remote sensing satellite ERS were used by Quegan *et al.* [2000] for forest mapping by analyzing the temporal stability of forest. Further, McCloy & Lucht [2004] developed and tested a method for comparing the complex spatio-temporal patterns in two long ERS time series data of the seasonal cycles of vegetation for a large part of the global land surface. ERS inverse SAR (InSAR) for land-cover classification was investigated and validated by Engdahl & Hyyppa [2003].

A hierarchical Bayesian modeling of Satellite Image Time Series (SITS) information content was proposed by Heas & Datcu [2005] to learn and retrieve spatio-temporal structures, which enables us to link the interest of a user to specific spatio-temporal structures. This method is composed of two inference steps: an unsupervised modeling of dynamic clusters resulting in a graph of trajectories; and an interactive learning procedure based on graphs which leads to the semantic labeling of spatio-temporal structures. Through this approach, temporal patterns can be indexed by the trajectories.

Two years later, the information-bottleneck principle was applied by Gueguen & Datcu [2007] to extract relevant information from SITS. GMRF and auto-binomial Gibbs random field models were applied to characterize the spatio-temporal structures. The information-bottleneck principle was applied to cluster the spatio-temporal parameters characterizing the evolution patterns, where the optimal number of clusters is determined by a rate-distortion approach. After clustering, the evolution classes can be indexed and discovered. The same authors Gueguen & Datcu [2008] applied spatio-temporal pattern indexing based on compression to SITS mining. Normalized Sufficient Compression Distance (NSCD), an extension of the Normalized Compression Distance (NCD) proposed by Li *et al.* [2004] was proposed to assess the quantity of relevant information that an object provides about another. Based on the compression of spatio-temporal patterns, an indexing schema was used to extract relevant information from the SITS.

A frequent sequential pattern extraction method was proposed by Julea *et al.* [2011] to extract temporal evolutions at the pixel level. As the number of evolution classes increases exponentially with respect to the quantization levels, pixel brightness quantization was applied to reduce the range of classes at each time step. Otherwise, the computational burden would increase greatly. Furthermore, local context is not considered.

Dynamic time warping was proposed by Petitjean *et al.* [2012] to solve the two problems in analyzing high temporal resolution satellite image series, namely irregular sampling in the temporal dimension and the need for comparing pairs of time series with different numbers of samples. A classification of optical images validated the ability of time warping as a similarity measure.

A novel unsupervised algorithm, called CLimate and rEmote sensing Association patteRns Miner (CLEARMiner), was developed by Romani *et al.* [2013] for mining association patterns in climate and remote sensing time series. It includes a symbolic representation of SITS in order to identify patterns in multi-temporal satellite images and associates them with patterns in other series within a temporal sliding window. A system that implements this method was developed for the monitoring of sugar cane fields.

To analyze land cover evolution, Latent Dirichlet Allocation (LDA) was applied by Vaduva *et al.* [2013] to a change map time series, which was computed from a SITS using four similarity measures. The authors show that the LDA model enables us to discover a wide range of scene evolution classes derived from the dynamic behavior of the land cover. A major drawback of this method is the lack of reliable ground truth related to the dynamic evolution of the land cover.

Multi-temporal TerraSAR-X images and continuous Global Positioning System (GPS) measurements were used by Fallourd *et al.* [2011] to monitor moderate glacier displacements. The results obtained with four time series covering the French Chamonix Mont-Blanc glaciers over one year show that the SAR phase information is rarely preserved after 11 days on such glaciers, whereas the high resolution brightness information allows the main glacier features to be observed and displacement fields on the textured areas to be derived. Similarly, a time series of ERS SAR images acquired during 1996 and 2005 was used by Chen *et al.* [2011] to monitor and analyze the coast line changes on Western Taiwan. Time series of TerraSAR-X dual-polarization images were applied by Lopez-Sanchez *et al.* [2011] to monitor the rice growing status in Spain. A time series of COSMO-SkyMed (CSK) images is exploited in Notarnicola *et al.* [2013] for the detection of seasonal snow cover in alpine areas.

Image time series of the moderate-resolution (optical) imaging spectroradiometer MODIS have been widely used for land cover monitoring. A method for unsupervised land cover change detection was presented in Salmon *et al.* [2011], which operates on short-time Fourier transform coefficients computed over subsequences of 8-day composite MODIS reflectance data that were extracted with a temporal sliding window. MODIS image time series were also applied by Grobler *et al.* [2012, 2013] for land cover classification and for the mapping of burned areas by Bastarrika *et al.* [2011]. An extended Kalman filter for MODIS NDVI time series data was proposed in Kleynhans *et al.* [2010] to improve land cover class separation.

A software tool was developed by Udelhoven [2011] for the retrieval of temporal patterns from global satellite archives, such as MODIS, AVHRR, MERIS and SPOT-Vegetation Satellites [2014]. Function fitting to NDVI time series was used by Jonsson & Eklundh [2002] to extract seasonal patterns. A Fourier regression algorithm on the NDVI time series of Landsat images was proposed by Brooks *et al.* [2012] to recover the missing values. A binary coding method for Landsat image time series was used by Lee [2008] to map deforestation and the age of evergreen trees, which is actually a post-classification change detection method. Multi-temporal satellite images and ancillary data were used to monitor landscape changes in LANDFIRE Vogelmann

et al. [2011], which is a large interagency project designed to provide nationwide spatial data for fire management applications.

InSAR time series for subsidence monitoring is also a branch in Earth observation. An InSAR method for persistent scatterers based on a freely connected network for subsidence detection was presented by Liu *et al.* [2011a] using a time series of TerraSAR-X SAR images. A joint analysis of InSAR time series using persistent scatterers and short baseline subset data was presented in Yan *et al.* [2012] to monitor the subsidence in Mexico City.

3.5 Image Information Mining

For years, interactive systems which allow the discovery of hidden patterns and the indexing of high dimensional datasets through visual content mining have been continuously developed. Typical implementations that approach these issues are the Knowledge-driven Information Mining (KIM) system of Datcu & Seidel [2005] and the Geospatial Information Retrieval and Indexing (GeoIRIS) system of Shyu *et al.* [2007]. KIM is an interactive system based on human-centered concepts, which allows the user to guide the interactive learning process, while the system continuously gives the relevance feedback about the performed training actions and searches the archive for relevant images. GeoIRIS is a content-based multi-modal geospatial information retrieval and indexing system, which allows automatic feature extraction, visual content mining from large-scale image databases, and high-dimensional database indexing for fast retrieval. These two interactive systems have achieved great success and played an important role in satellite image information mining.

However, as pointed out by Popescu *et al.* [2012], the increased spatial resolution of modern spaceborne SAR sensors renders the pixel level processing insufficient as targets are not any more observed in isolation; instead, groups of objects, e.g., houses, bridges, roads, etc., need to be recognized in their spatial context. Therefore, high resolution SAR image information retrieval and mining has to be performed at a higher patch level. Nevertheless, the high volume of our databases, especially of the multi-temporal databases, makes interactive information mining very demanding from a computational point of view. The interactive learning process should be fast enough such that the users can receive the system feedback within a few seconds after selecting the training samples. There already exist some methods and systems to approach this computational issue, like various kinds of coarse-to-fine strategies to reduce the data volume to be processed in each interaction. The most notable one is the face detector proposed by Viola & Jones [2004], which is based on the fact that faces are scarce in images. Thus, it is completely a waste of effort to classify the non-face regions. To discard non-face regions as early as possible, a cascade classifier was proposed, which has achieved impressive results and speeded up the face detection.

Although a coarse-to-fine strategy can significantly reduce the computational effort, it is still not easy to obtain a sufficient number of training samples for a large scale satellite image database due to limited financial resources and the large manual effort. To overcome the shortage of training samples, both active learning Tuia *et al.* [2011] and semi-supervised learning Zhu *et al.* [2009] have attracted much attention. Semi-supervised learning is a learning strategy between unsupervised and supervised learning, which uses both labeled and unlabeled data for learning as the labeled data is usually quite rare. On the other hand, active learning iteratively selects the most informative samples for manual labeling in order to optimize the decision surface. The key component in active learning is the sampling strategy.

In the case of a large dataset, each class covers only a limited part in the image, thus it is a waste of effort to classify irrelevant patches. Taking advantage of both cascade classification and

active learning, a cascade active learning method was proposed for object retrieval by Blanchart *et al.* [2011], which has shown promising ability in terms of both accuracy and speed. This method has been adapted to SAR image annotation by Cui & Datcu [2012]. In chapter 6, we continue this line of work and present our new contributions for interactive spatial and temporal SAR image information mining. An extended cascade active learning system has been developed for patch level spatial and temporal SAR image information mining. This system can be used as a tool to generate ground truth through interactive learning, which is of significant practical importance.

3.6 Dimension Reduction for Visualization of Multi-Temporal SAR Images

Conventionally, visualization is not a task of static image information mining. However, the story becomes different for multi-temporal image information mining because it is not trivial to visualize SAR ITS data. The same issue exists for multispectral and hyperspectral image interpretation Bajcsy & Groves [2004]. The conventional solutions are dimension reduction techniques, which represent a well-established research topic and many methods are available. The most common method is Principal Component Analysis (PCA) Du *et al.* [2008]; Tyo *et al.* [2003], which projects each image to a sequence of k orthogonal components such that the variance of the data is maintained as much as possible. In addition, various information measures have been applied to band selection for visualization Le Moan *et al.* [2011]. Due to new developments in dimension reduction, especially in manifold learning, such as Isometric Mapping (ISOMAP) Tenenbaum [2000] and Locally Linear Embedding (LLE) Roweis & Saul [2000], dimension reduction while preserving the pairwise distances has been developed. This includes hyperspectral image visualization by Bachmann *et al.* [2006]; Cui *et al.* [2009]. A relevant topic is image fusion, which integrates all the information of multi-temporal images and generates a new representation of them. A linear fusion of image sets for visualization was proposed by Jacobson *et al.* [2007] by projecting them onto basis functions. Convex optimization was proposed for visualization by Cui *et al.* [2009] with the constraints of preserving the pairwise distances and maintaining the discriminability of pixels with different spectral signatures. Similarly, an optimization-based approach was proposed in Kotwal & Chaudhuri [2012] by optimizing a multi-objective cost function under given constraints on the fused image. More recently, Minimum-Redundancy-Maximum-Relevance (MRMR) criterion was applied by Bratasanu *et al.* [2012] to band discovery for the exploratory visual analysis of satellite images. One of the biggest problems in applying these methods is that they distort the information, which means the users cannot easily recognize the content from the representation after dimension reduction. Thus, it is hard to visually interpret the resulting pseudo color image. Furthermore, all these methods lose a certain amount of information and not all information is visible to the human eye. Therefore, in this thesis, we rely on a simple animation representation of SAR ITS Meisner *et al.* [1999].

3.7 Conclusion and Proposed Concepts

Based on the review which was just presented, it is evident that most research activities in this field are focused on the pixel level, which becomes inadequate for VHR SAR images. Furthermore, the volume of the involved datasets is small, thus a conclusion drawn from the evaluation of a small dataset may be not reliable. With a large database of satellite images for real applications, it is not trivial to know what patterns and how many patterns exist. This is the

most important and practical issue and makes the automated generation of ground truth data infeasible. Developing methods and systems for exploring large scale databases is of significant practical importance. In this thesis, we propose the following two concepts to address these issues.

3.7.1 Patch Level Spatial and Temporal SAR Image Characterization

Due to the intrinsic drawbacks of pixel level methods for high resolution SAR images, we propose to work on the patch level for both feature extraction and learning. For high resolution SAR images, the assumption of stationary texture is no longer valid. A typical characteristic of VHR SAR images is the appearance of bright spots, especially in urban areas. Thus, the results reached by pixel level classification are not meaningful, because some bright pixels in SAR images represent strong scatterers due to single reflection or double bouncing, and they do not describe a valid physical class. From the feature extraction point of view, features extracted from a single pixel do not provide a valid interpretation of the local neighborhood. In the worst case, the information probably characterizes something else because of the coherent SAR imaging mechanism. Without prior knowledge and consideration of the complex structure arrangement in the local context, a scene categorization cannot be achieved. In this sense, only a group of pixels can provide a clear meaning and can possibly be interpreted. If a large context is considered, the semantic gap between the signal and the content can be probably narrowed in interpreting high resolution SAR images. Conventional SAR features mainly refer to textures. In the case of high resolution SAR images, texture becomes less important than context. In addition, texture features become less descriptive for urban areas. To solve this problem, we propose a new perspective on patch level feature extraction based on the Bag-of-Words method, which shows strong discriminative capabilities for very high resolution SAR images.

3.7.2 Cascade Active Learning for Spatial and Temporal SAR Image Information Mining

Conventional techniques, such as classification and segmentation, require strong prior knowledge of the data set, such that it is feasible to generate ground truth data, although this may be expensive. Thus, it is not possible to assess the resulting classification accuracy without reference data, which is rarely available in practice. Content-based indexing would be a candidate solution for this problem, which is the strategy we propose in this thesis. In this case, the relevant information will be returned to the user by giving examples, which is very useful in practical applications. On the other hand, the volume of the EO data is becoming quite huge. Efficient learning methods have to be developed for fast indexing and the discovery of hidden temporal patterns in SAR ITS. In most cases, the relevant class in an image covers only a small part of it. It is not necessary to train and learn on irrelevant patches, which consumes a lot of computational effort. Based on this schema, a cascade active learning method relying on a coarse-to-fine strategy is developed for spatial and temporal SAR image information mining. With this method, relevant patches of a certain temporal pattern can be discovered quickly.

Chapter 4

Information Similarity Metrics and Estimation for Multi-Temporal SAR Image Analysis

Since the last decade, information similarity measures have gained great interest in many disciplines. In recent years, information similarity measures have been applied to multi-temporal image analysis, especially to change detection and similarity assessments. The information similarity measures presented in this chapter are based on the basic concept of entropy given by Shannon [1948] in his prestigious paper, which has laid the foundation for the discipline of information theory.

In this chapter, we first briefly recall statistical models for SAR images in section 4.1, which are needed to estimate information measures. We focus on several important ones derived from either the product model or the speckle model. Then we present practical estimation methods for high resolution SAR images, followed by an evaluation using a heterogeneous dataset. In section 4.2, information similarity metrics are introduced. Based on the statistical models and the estimation methods, information similarity measures are applied to unsupervised SAR change detection. In order to conduct an objective evaluation and comparison, a benchmark dataset was generated as outlined in section 4.3 by simulating various kinds of changes. A comprehensive evaluation of information similarity measures for SAR image change detection is carried out. Due to the statistical similarity between SAR images and wavelet coefficients, statistical models are applied and evaluated in the wavelet domain for SAR change detection. This will be described in section 4.4.

4.1 SAR Statistical Models and Estimation

From a methodological point of view, SAR statistical models fall into three categories: nonparametric, semi-parametric, and parametric approaches. Nonparametric methods make no assumptions about the form of the distribution. There are three typical nonparametric estimation methods, namely histogram analysis, kernel density estimation and nearest neighbor estimation. Since the histogram method has two drawbacks: discontinuity and scaling with dimensionality, a kernel density estimation is preferable for joint density estimation Bishop [2006]. Semi-parametric methods mainly refer to finite mixture models. Although the semi-parametric methods have some merits, it is difficult to determine the number of mixture model components and the initial values of the involved parameters. Additionally, we face some computational

problems, such as ill-conditioned covariance matrices. Therefore, these kind of methods are usually not practical in modeling the joint distribution of multi-temporal images. Due to their physical principle, parametric SAR models are preferred by the SAR community. A large variety of parametric models have been proposed. There are three main kinds of parametric SAR models: models derived from the speckle model, models based on the product model, and empirical ones Li *et al.* [2010]. In this section, we focus on parametric SAR models and, for the sake of completeness, nonparametric and semi-parametric models are briefly reviewed.

4.1.1 SAR Speckle Model

The first kind of models are based on the speckle model, which assumes that each resolution cell contains a sufficient number of statistically independent scatterers. A reflected radar echo is the sum of the reflections from each scatterer, as shown in Eq. (4.1)

$$Z = X + Yi = Ae^{i\phi} = \sum_{k=1}^{N} A_k e^{i\phi_k} \tag{4.1}$$

where X and Y are the real and imaginary part of a complex-valued SAR image pixel, A and θ are the amplitude and the phase of the reflected signal Z, N is the number of the scatterers and A_k and θ_k are the amplitude and phase of the k^{th} scatterer. To find the statistical distribution of the amplitude A, we need to model both the real $X = A\cos\phi$ and imaginary components $Y = A\sin\phi$. Based on the assumption that each resolution cell contains a sufficient number of statistically independent scatterers, both components are independent identically distributed Gaussian random variables with zero-mean and a variance of $\sigma/2$ (σ is the radar cross section (RCS)) due to the central limit theorem Oliver & Quegan [1998]. Thus, the joint Probability Density Function (PDF) is

$$p_{X,Y}(x,y) = p_X(x)p_Y(y) = \frac{1}{\pi\sigma}\exp\Big(-\frac{x^2+y^2}{\sigma}\Big). \tag{4.2}$$

Accordingly, single look amplitude data $A = \sqrt{X^2+Y^2}$ should have a Rayleigh distribution and thus the intensity $I = A^2$ follows a negative exponential distribution [1], defined respectively in Eq. (4.3) and Eq. (4.4) [2].

$$p_A(x) = \frac{2x}{\sigma}\exp\Big(-\frac{x^2}{\sigma}\Big), \quad x \geq 0 \tag{4.3}$$

$$p_I(x) = \frac{1}{\sigma}\exp\Big(-\frac{x}{\sigma}\Big), \quad x \geq 0 \tag{4.4}$$

In the case of multi-look SAR images, the intensity is the average of all looks. Formally, the L-look average intensity is given by Eq. (4.5)

$$I = \frac{1}{L}\sum_{k=1}^{L} I_k, \quad L \geq 0 \tag{4.5}$$

It is known that the L look intensity I has a Gamma distribution, given in Eq. (4.6)

$$p_I(x) = \frac{1}{\Gamma(L)}\Big(\frac{L}{\sigma}\Big)^L x^{L-1}\exp\Big(-\frac{Lx}{\sigma}\Big), \quad x \geq 0 \tag{4.6}$$

[1] The distribution of SAR amplitude can be derived by the transform relation to intensity $f_A(x) = 2xf_I(x^2)$

[2] Throughout this thesis, intensity variables shall be denoted with subscript "I", and amplitude variables with subscript "A".

where L is the number of looks and $\Gamma(x)$ is the Gamma function. The i^{th} order raw moment is shown in Eq. (4.7).

$$m_i = \frac{\Gamma(i+L)}{\Gamma(L)}\Big(\frac{\sigma}{L}\Big)^i \tag{4.7}$$

In particular, the mean value of the intensity image is σ and its variance is σ^2/L. From this, we can see that there is a fixed relation between mean and variance. Accordingly, the amplitude follows a square root Gamma distribution as shown in Eq. (4.8)

$$p_A(x) = \frac{2}{\Gamma(L)}\Big(\frac{L}{\sigma}\Big)^L x^{2L-1}\exp\Big(-\frac{Lx^2}{\sigma}\Big), \quad x \geq 0 \tag{4.8}$$

The derivation of all these models is based on the Gaussian assumption and the independence of both real and imaginary components. However, in case of increased resolution, non-Gaussian speckle has been observed frequently. This is demonstrated in Fig. 4.1. The two images in the first row are respectively the real and imaginary components of a high resolution TerraSAR-X image. In the second row, two Gaussian PDFs are estimated from the two components. It is evident that, there is a large discrepancy between the two histograms and the estimated density functions. On the other hand, if we fit a Generalized Gaussian Distribution (GGD) with a shape parameter γ, a scale parameter λ, and a zero mean parameter $m = 0$, shown in Eq. (4.9), for the two components, the discrepancy decreases significantly. This is the basic idea of a Generalized Gaussian Rayleigh (GGR) distribution proposed by Moser *et al.* [2006]. The amplitude PDF of a GGR is shown in Eq. (4.10).

$$p_X(x) = \frac{\gamma}{2\lambda\Gamma(\lambda)}\exp\Big(-|\gamma(x-m)|\Big), \quad \lambda,\gamma > 0 \tag{4.9}$$

$$p_A(x) = \frac{\gamma^2 x}{\lambda^2\Gamma^2(\lambda)}\int_0^{\frac{\pi}{2}}\exp\Big(-(x\gamma)^{\frac{1}{\lambda}}(|\cos\theta|^{\frac{1}{\lambda}} + |\sin\theta|^{\frac{1}{\lambda}})\Big)d\theta, \quad \lambda,\gamma > 0, x \geq 0 \tag{4.10}$$

It can be seen from Eq. (4.10) that the integral cannot be computed analytically; thus, it has very bad analytical properties. In addition to GGD, the zero-mean symmetric α-stable distribution was proposed by Kuruoglu & Zerubia [2004] for modeling the real and imaginary components and has resulted in a generalization of the Rayleigh distribution, called heavy-tailed Rayleigh distribution. The PDF of a heavy-tailed Rayleigh distribution is given by Eq. (4.11).

$$p_A(x) = x\int_0^{+\infty}\rho\exp(-\gamma\rho^\alpha)J_0(x\rho)d\rho, \quad \alpha,\gamma > 0, x \geq 0 \tag{4.11}$$

where $J_0(x)$ is the zeroth order Bessel function of the first kind, α is a stability parameter, and γ is a skewness parameter. Similarly, the heavy-tailed Rayleigh distribution does not have good analytical property, which complicates the parameter estimation and restricts its applicability. Recently, a two-side Generalized Gamma Distribution (GΓD) was applied by Li *et al.* [2010] for modeling the complex-valued components, which gives a Generalized Gamma Rayleigh (GΓR) distribution. The two-sided GΓD is defined in Eq. (4.12)

$$p_A(x) = \frac{\nu}{2\eta\Gamma(\kappa)}\Big(\frac{|x|}{\eta}\Big)^{\kappa\nu-1}\exp\Big(\frac{|x|}{\eta}\Big)^\nu, \quad \nu,\kappa,\eta > 0 \tag{4.12}$$

where ν, κ, and η are the power, shape, and scale parameters. Based on the assumption that both real and imaginary components follow a two-sided GΓD, the PDF of the amplitudes can be

written as

$$p_A(x) = \Big(\frac{\nu}{\eta^{\kappa\nu}\Gamma(\kappa)}\Big)^2 x^{2\kappa\nu-1}\int_0^{\frac{\pi}{2}} |\cos\theta\sin\theta|^{\kappa\nu-1}\exp\Big(-(\frac{x}{\eta})^{\nu}(|\cos\theta|^{\nu}+|\sin\theta|^{\nu})\Big)d\theta \tag{4.13}$$

where the parameters ν, κ, and η are the same as those in GΓD. Although all these models have a solid theoretical foundation, parameter estimation has to rely on numerical methods that can decrease the accuracy and increase the computing time. In addition, it has been demonstrated by Li *et al.* [2011] that the one-sided GΓD, presented in section (4.1.3), can achieve competitive performance in addition to its simple parameter estimation. We conclude that simple models, which can be easily estimated with high accuracy, are preferable in practical applications.

4.1.2 SAR Product Model

An interesting property of Eq. (4.6) is that it can be interpreted as the product of a fixed RCS σ with a multiplicative noise process with a unit mean Gamma distribution, which leads to the well-known product model. The product model states that the reflected radar echo is the product of an underlying RCS σ with a multiplicative speckle n_I. Then, the multi-look intensity I can be expressed as the product $I = \sigma n_I$, where the speckle intensity follows the unit mean Gamma distribution shown in Eq. (4.14).

$$p_{n_I}(x) = \frac{L^L x^{L-1}}{\Gamma(L)}\exp(-Lx), \quad L \geq 0, x \geq 0 \tag{4.14}$$

where L is the number of looks. It can also be proven mathematically that the product model holds for both the amplitude and the complex echo, with the relation $A = \sqrt{\frac{\sigma}{2}}n_A$ and $Z = \sqrt{\frac{\sigma}{2}}n_Z$.

The product model has played an important role in developing SAR image models because it can separate the imaging process into two independent random processes. In order to model SAR images, we need to find models for both RCS and speckle. If the reflecting surface is homogeneous, the assumption of a constant RCS is valid, which leads to a square root Gamma distribution of SAR amplitudes. In the case of a heterogenous region, an appropriate RCS model has to be developed. Studies of sea clutter by Ward [1981] showed that the underlying RCS can be well modeled by a square root Gamma distribution. Together with the square root Gamma distributed speckle amplitude, the SAR amplitude has a $\mathcal{K}$-root distribution, defined in Eq. (4.15).

$$p_A(x) = \frac{2\lambda L}{\Gamma(L)\Gamma(\alpha)}\Big(\lambda L x\Big)^{\frac{L+\alpha}{2}-1}K_{\alpha-L}(2\sqrt{\lambda L x}) \tag{4.15}$$

where λ is a scale parameter, α is an order parameter, L is the number of looks, and $K_n(\cdot)$ is the n^{th} order modified Bessel function of the second kind.

The i^{th} order raw moment of the $\mathcal{K}$ distribution is given as Eq. (4.16), which can be used for parameter estimation by the method of moments (See section 4.1.4.1).

$$E(I^i) = (\lambda L)^{-i}\frac{\Gamma(\alpha+i)\Gamma(L+i)}{\Gamma(\alpha)\Gamma(L)} \tag{4.16}$$

It can be proven that the cumulative distribution function (CDF) of a $\mathcal{K}$ random variable is the formula shown in Eq. (4.17), where ${}_1F_2$ is the generalized hypergeometric function. The proof is presented in the appendix

$$\begin{aligned} F_A^{\mathcal{G}^0}(x) = &\frac{\Gamma(\alpha-L)}{L\Gamma(L)\Gamma(\alpha)}(\lambda L x)^L {}_1F2(L; 1+L-\alpha, L+1; \lambda L x) \\ &+ \frac{\Gamma(L-\alpha)}{\alpha\Gamma(L)\Gamma(\alpha)}(\lambda L x)^{\alpha} {}_1F2(\alpha; 1+\alpha-L, \alpha+1; \lambda L x) \end{aligned} \tag{4.17}$$

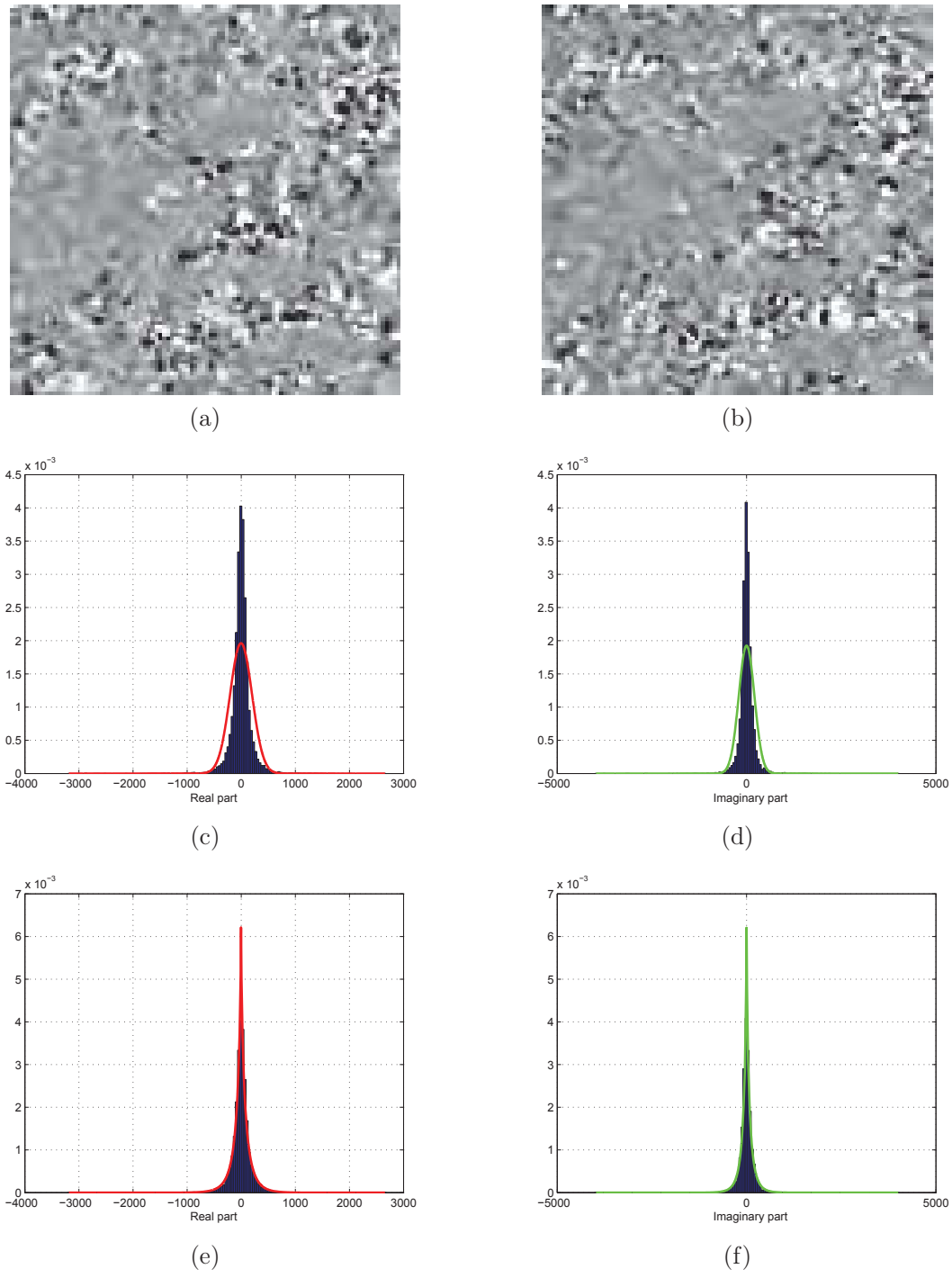


Figure 4.1: An example of non-Gaussian speckle in high resolution SAR images: (a)(b) the real and imaginary components; (c)(d) Gaussian fitting of the two component histograms; (e)(f) Generalized Gaussian fitting of the two component histograms.

However, as pointed out by Gu & Abraham [2001], although an analytical form of the CDF of a $\mathcal{K}$ distribution is available, it is still not trivial to compute it because the generalized hypergeometric function ${}_1F_2(x)$ goes to infinity as x increases. It is necessary to develop an approximation method for computing the CDF of a $\mathcal{K}$ distribution. As observed by Lopes *et al.* [1990] and Ulaby *et al.* [1986], a $\mathcal{K}$ distribution cannot properly model extremely heterogeneous regions, like urban areas. Therefore, Frery *et al.* [1997] proposed the $\mathcal{G}^0$ distribution for extremely heterogeneous regions in high resolution SAR images by assuming that the RCS follows the reciprocal of a square root Gamma distribution. The $\mathcal{G}^0$ model of the SAR intensity is defined as

$$p_I(x) = \frac{L^L\Gamma(L-\alpha)}{\gamma^\alpha\Gamma(L)\Gamma(-\alpha)}\frac{x^{L-1}}{(\gamma+Lx)^{L-\alpha}}, \quad \gamma>0, \alpha<0 \tag{4.18}$$

where L is the number of looks, γ is a scale parameter and α is a shape parameter. The i^{th} order raw moment is given by Eq. (4.19).

$$E(I^i) = \left(\frac{\gamma}{L}\right)^i \frac{\Gamma(L+i)\Gamma(-\alpha-i)}{\Gamma(L)\Gamma(-\alpha)} \tag{4.19}$$

The corresponding cumulative distribution function for SAR amplitude data is given as follows (which will be used for change simulation and Kolmogorov-Smirnov tests).

$$F_A(x) = F_{2L,-2\alpha}\Big(-\frac{\alpha x^2}{\gamma}\Big) \tag{4.20}$$

where F_{v_1,v_2} is the F cumulative distribution function Johnson [1995], i.e.:

$$F_{v_1,v_2}(x) = \frac{\Gamma\left(\frac{v_1+v_2}{2}\right)}{\Gamma\left(\frac{v_1}{2}\right)\Gamma\left(\frac{v_2}{2}\right)}\left(\frac{v_1}{v_2}\right)^{\frac{v_1}{2}}\int_0^x t^{\frac{v_1-2}{2}}\left(1+\frac{v_1}{v_2}t\right)^{-\frac{v_1+v_2}{2}}dt \tag{4.21}$$

It is worth noting that the $\mathcal{G}^0$ distribution is completely equivalent to the Fisher distribution Tison *et al.* [2004] defined in Eq.(4.22)

$$p_I(x) = \frac{\Gamma(L+M)}{\Gamma(L)\Gamma(M)}\frac{L}{M\mu}\frac{\left(\frac{Lx}{M\mu}\right)^{L-1}}{\left(1+\frac{Lx}{M\mu}\right)^{L+M}} \tag{4.22}$$

By equalizing the two functions, we can obtain the following relations between the two sets of parameters:

$$L=n, \quad M=-\alpha, \quad \mu=-\gamma/\alpha. \tag{4.23}$$

4.1.3 Empirical Models

A number of empirical models have been derived based on the experimental analysis of SAR images, like the log-normal distribution Oliver & Quegan [1998], the Weibull distribution Oliver [1993], the Fisher distribution Tison *et al.* [2004], or the one-sided GΓD by Li *et al.* [2011]. In this section, we focus on the GΓD for SAR image modeling because it has an efficient approximate parameter estimation method. The PDF of the GΓD described by Krylov & Zerubia [2010]; Li *et al.* [2007, 2011] [1] is presented in Eq. (4.24)

$$p_A(x) = \frac{\beta x^{\beta\lambda-1}}{\alpha^{\beta\lambda}\Gamma(\lambda)}\exp\Big(-(\frac{x}{\alpha})^\beta\Big) \tag{4.24}$$

[1] The two versions differ slightly, but they follow the same mathematical rules.

where α is the scale parameter, β is the shape parameter and λ is the index shape parameter. The corresponding CDF of a GΓD is given by Eq. (4.25)

$$F_A(x) = \frac{\gamma\left(\lambda, (\frac{x}{\alpha})^\beta\right)}{\Gamma(\lambda)} \tag{4.25}$$

where γ denotes the lower incomplete gamma function. Based on the Method of Log-Cumulants (MoLC) (See section 4.1.4.3), the equations to be solved for the parameters are shown in Eq. (4.26).

$$\kappa_0 = \ln(\alpha) + \frac{\Psi(0,\kappa)}{\beta} \quad \kappa_i = \frac{\Psi(i-1,\kappa)}{\beta^i}, \quad i = 2, 3, ... \tag{4.26}$$

After substituting the following asymptotic decomposition of the polygamma functions,

$$\begin{aligned} \Phi(1,x) &= x^{-1} + 0.5x^{-2} + O(x^{-2}) \\ \Phi(2,x) &= -x^{-2} - x^{-3} + O(x^{-3}) \end{aligned} \tag{4.27}$$

the following closed expression can be obtained

$$\begin{aligned} \hat{\beta} &= \sqrt{\frac{\Psi(1,\hat{\kappa})}{\kappa_2}} \quad \hat{\alpha} = \exp\left(\kappa_1 - \frac{\Psi(0,\hat{\kappa})}{\hat{\beta}}\right) \\ \hat{\lambda} &= -\frac{a_2}{3a_3} + \sqrt[3]{\sqrt{\frac{p^2}{4} + \frac{q^3}{27}} - \frac{p}{2}} + \sqrt[3]{-\frac{p}{2} - \sqrt{\frac{p^2}{4} + \frac{q^3}{27}}} \end{aligned} \tag{4.28}$$

where

$$\begin{aligned} a_0 &= \kappa_3^2 - 8\kappa_2^3 \quad a_1 = 6\kappa_3^2 - 16\kappa_2^3 \quad a_2 = 12\kappa_3^2 - 8\kappa_2^3 \\ a_3 &= 8\kappa_3^2 \quad p = \frac{27a_3^2a_0 - 9a_1a_2a_3 + 2a_2^3}{27a_3^3} \quad q = \frac{a_1}{a_3} - \frac{a_2^2}{3a_3^2} \end{aligned} \tag{4.29}$$

Therefore, a closed form expression of each parameter is available, which can significantly speed up the estimation process and make it applicable to practical applications.

4.1.4 Parameter Estimation

In this section, conventional parameter estimation methods, such as the method of moments and maximum likelihood estimation are briefly reviewed. Then we focus on MoLC, which has shown promising ability in estimating the parameters that govern the distribution of positive random variables Nicolas [2002].

4.1.4.1 Method of Moments (MoM)

The method of moments is a technique to estimate the parameters $\theta = (\theta_1, \theta_2, ..., \theta_n)$ governing a PDF $p_X(x;\theta)$ of a random variable X. Given the samples drawn independently from $p_X(x;\theta)$, the θ parameter can be estimated by matching the first n raw moments $m_i(\theta)$ to the corresponding n empirical moments $\hat{\mu}_i$ and solving the resulting system of simultaneous equations:

$$\begin{aligned} \hat{\mu}_1 &\equiv E[X^1] = m_1(\theta), \\ \hat{\mu}_2 &\equiv E[X^2] = m_2(\theta), \\ &\vdots \\ \hat{\mu}_n &\equiv E[X^n] = m_n(\theta) \end{aligned} \tag{4.30}$$

There are two critical issues to consider when we apply the method of moments to parameter estimation. First, there must be a closed form expression of raw moments $m_i(\theta)$; otherwise, we cannot construct the equations. Second, we need an efficient method to solve the equations. If the system of equations has multiple roots, one has to isolate the correct one, like in the case of GΓD described by Wingo [1987]. In the case of an intensity distribution shown in Eq. (4.6), the two parameters, i.e., σ and L, can be estimated as shown in Eq. (4.31).

$$\sigma = \hat{\mu}_1, \quad L = \frac{\hat{\mu}_1^2}{\hat{\mu}_2} \tag{4.31}$$

In this case, there exists a closed form expression thanks to the good mathematical properties. Nevertheless, the accuracy of estimation is usually very low compared with Maximum Likelihood Estimation (MLE) that will be described in the next section. In practice, the MoM is used for the initialization of the MLE because it has the advantage of simplicity.

4.1.4.2 Maximum Likelihood Estimation (MLE)

Given the samples drawn independently from a distribution $p_X(x;\theta)$ governed by $\theta = (\theta_1, \theta_2, ..., \theta_n)$, MLE determines the parameters, with which the samples had been generated with maximum probability. Under the assumption that the samples are independent and identically distributed, the probability of the samples is given by the product of probability of each sample, that is Eq. (4.32).

$$\mathcal{L}(\theta \,|\, x_1, \ldots, x_n) = p_X(x_1, x_2, \ldots, x_n \mid \theta) = \prod_{i=1}^{n} P(x_i|\theta). \tag{4.32}$$

In the case of a Gamma distribution, already defined in Eq. (4.6), the log-likelihood function can be written as

$$\mathcal{L}(\theta \,|\, x_1, \ldots, x_n) = -n \ln \Gamma(L) + nL \ln \frac{L}{\sigma} + (L-1) \sum_{i=1}^{n} \log x_i - \frac{L}{\sigma} \sum_{i=1}^{n} x_i \tag{4.33}$$

Taking the derivative with respect to σ and setting it to zero yields the estimation of $\hat{\sigma}$ as follows

$$\hat{\sigma} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{4.34}$$

Note that the estimation of σ corresponds to the method of moments. Substituting this estimate of σ into the log-likelihood function gives

$$\mathcal{L}(\theta \,|\, x_1, \ldots, x_n) = -n \ln \Gamma(L) + nL \ln L - nL \ln \frac{\sum_{i=1}^{n} x_i}{n} + (L-1) \sum_{i=1}^{n} \ln x_i - nL \tag{4.35}$$

Taking the derivative with respect to L and setting it to zero gives the equation

$$\ln L - \psi(L) = \ln \left(\frac{\sum_{i=1}^{n} x_i}{n} \right) - \frac{\sum_{i=1}^{n} \ln x_i}{n} \tag{4.36}$$

where $\psi(L) = \frac{\Gamma' Lk)}{\Gamma(L)}$ is the digamma function. This equation can be solved efficiently using numerical methods. Although this is not very complex, it would become very difficult for more complicated cases, like the $\mathcal{K}$ distribution or the $\mathcal{G}^0$ distribution, where the parameter estimation becomes an issue of hard optimization. In addition, a theoretical proof of the existence and uniqueness of the solution is no trivial problem.

4.1.4.3 Method of Log-Cumulants (MoLC)

Proposed by Nicolas [2002], MoLC has been introduced to SAR data analysis for image distribution estimation. It uses a Mellin transform instead of a Fourier transform to analyze random variables defined over $\mathbf{R}^+$. A set of parallel concepts, called second kind statistics, are defined in contrast to conventional ones, where one applies a moment-generating function (MGF) and a cumulant-generating function (CGF). The MoLC second kind moment generating function is defined as the Mellin transform of a probability density function $p_X(x)$, shown in Eq. (4.37)

$$\phi_x(s) = \int_0^{+\infty} x^{s-1} p_X(x) dx \tag{4.37}$$

By analogy, the i^{th} order second kind moment is defined as the i^{th} order derivative of the MGF evaluated at $s = 1$.

$$m_i(s) = \frac{d^i \phi_x(s)}{ds^i}\bigg|_{s=1} = \int_0^{\infty} (\log x)^i p_X(x) dx \tag{4.38}$$

The CGF is defined in the same way as the natural logarithm of the second kind MGF, given in Eq. (4.39).

$$g_x(s) = \log \phi_x(s) \tag{4.39}$$

Similarly, the ith order second kind cumulant, also called log-cumulant, is given as the ith order derivative of CGF evaluated at $s = 1$

$$\kappa_i(s) = \frac{d^i g_x(s)}{ds^i}\bigg|_{s=1} \tag{4.40}$$

The main idea of MoLC for parameter estimation is the same for MoM, but applied to the second kind cumulant, rather than the raw moment. The second kind cumulant can be estimated by the samples $(x_1, x_2, ..., x_n)$ using

$$\hat{\kappa}_1 = \frac{1}{n}\sum_{i=1}^{n} \log x_i, \quad \hat{\kappa}_i = \frac{1}{n}\sum_{i=1}^{n} \Big(\log x_i - \hat{\kappa}_1\Big)^i \tag{4.41}$$

Therefore, parameter estimation is casted as solving equations. Following this method, we can derive the MoLC equations for the $\mathcal{G}^0$ distribution shown in Eq. (4.18) and for the $\mathcal{K}$ distribution shown in Eq. (4.15), which are shown respectively as Eq. (4.42) and Eq. (4.43). The MoLC equations for the Fisher distribution can be derived by the relation shown in Eq. (4.23).

$$\begin{aligned} &\ln(\gamma/L) + \psi(L) - \psi(-(\alpha)) = \hat{\kappa}_1 \\ &\psi(1, L) + \psi(1, -\alpha) = \hat{\kappa}_2 \\ &\psi(2, L) - \psi(2, -\alpha) = \hat{\kappa}_3 \end{aligned} \tag{4.42}$$

$$\begin{aligned} &\psi(L) + \psi(\alpha) - \ln(\lambda L) = \hat{\kappa}_1 \\ &\psi(1, L) + \psi(1, \alpha) = \hat{\kappa}_2 \\ &\psi(2, L) + \psi(2, \alpha) = \hat{\kappa}_3 \end{aligned} \tag{4.43}$$

4.1.5 Numerical Solution of MoLC

The straightforward method is to solve the MoLC equations for the parameters. However, there are some constraints on the parameters; therefore, we propose to cast solving the MoLC equations for the parameters as a constrained nonlinear least squares problem. We take a $\mathcal{G}^0$ distribution as an example for demonstration in this section.

4.1.5.1 Solving MoLC Equations

The Newton-Raphson method was applied by Galland *et al.* [2009] to solve the MoLC equations. Therefore, a Jacobian matrix, shown in Eq. (4.44), is required.

$$J(L,\alpha) = \begin{vmatrix} \psi(2,L) & -\psi(2,-\alpha) \\ \psi(3,L) & \psi(3,-\alpha) \end{vmatrix}. \tag{4.44}$$

The procedure iterates using Eq. (4.45) until reaching convergence.

$$\begin{pmatrix} L^{i+1} \\ \alpha^{i+1} \end{pmatrix} = \begin{pmatrix} L^{i} \\ \alpha^{i} \end{pmatrix} - J^{-1}(L^i,\alpha^i) \begin{pmatrix} \psi(1,L^i) + \psi(1,-\alpha^i) - \hat{\kappa}_2 \\ \psi(2,L^i) - \psi(2,-\alpha^i) - \hat{\kappa}_3 \end{pmatrix} \tag{4.45}$$

Although it has a quadratic convergence rate, it converges only when the initial values are quite close to the solution. Actually, this is not realistic in practical applications. The most critical issue in solving the equations is that there are constraints in the parameter space, i.e., $L > 0$ and $\alpha < 0$, which complicates the solution. Therefore, this is a problem of solving a constrained system of nonlinear equations, which cannot be approached by the conventional unconstrained Newton-Raphson method. In addition, the method is computationally very expensive because it needs to evaluate not only the function but also its derivative. This is a fatal drawback for sliding window type applications. Therefore, we need a better and robust solution.

4.1.5.2 Constrained Levenberg-Marquardt Nonlinear Minimization

Solving a constrained system of nonlinear equations is often casted as solving a constrained nonlinear least squares problem. The Levenberg-Marquardt method Madsen *et al.* [2004] is an iterative algorithm to minimize a function, which is the sum of squares of nonlinear real-valued functions. The Levenberg-Marquardt algorithm represents a damped Gauss-Newton method and has become the state-of-the-art algorithm for nonlinear least squares algorithms. Given a vector-valued function $\mathbf{F} : R^n \mapsto R^m$ with $m > n$, we would like to minimize its norm $f(\mathbf{x}) = \|\mathbf{F}(\mathbf{x})\|^2 = \frac{1}{2}\mathbf{F}(\mathbf{x})^T\mathbf{F}(\mathbf{x})$ equivalent to,

$$\mathbf{x} = \arg\min_{\mathbf{x}} \frac{1}{2}\mathbf{F}(\mathbf{x})^T\mathbf{F}(\mathbf{x}) \tag{4.46}$$

Based on a Taylor expansion, the function $\mathbf{F}(\mathbf{x})$ can be written as

$$\mathbf{F}(\mathbf{x}+\mathbf{h}) = \mathbf{F}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\mathbf{h} + O(\|h\|^2) \tag{4.47}$$

Therefore, the norm $f(\mathbf{x})$ can be written as

$$\begin{aligned} f(\mathbf{x}+\mathbf{h}) &= \frac{1}{2}\mathbf{F}^T\mathbf{F} + \mathbf{h}^T\mathbf{J}^T\mathbf{F} + \frac{1}{2}\mathbf{h}^T\mathbf{J}^T\mathbf{J}\mathbf{h} \\ &= f(\mathbf{x}) + \mathbf{h}^T\mathbf{J}^T\mathbf{F} + \frac{1}{2}\mathbf{h}^T\mathbf{J}^T\mathbf{J}\mathbf{h} \end{aligned} \tag{4.48}$$

Taking the derivative with respect to h gives

$$\frac{d\mathbf{f}(\mathbf{x}+\mathbf{h})}{d\mathbf{h}} = \mathbf{J}^T\mathbf{F} + \mathbf{J}^T\mathbf{J}\mathbf{h} \tag{4.49}$$

Because the Hessian matrix $H'' = \mathbf{J}^T\mathbf{J}$ is positive definite, there is a unique solution of Eq. (4.46), given by

$$(\mathbf{J}^T\mathbf{J})\mathbf{h_{gn}} = -\mathbf{J}^T\mathbf{F} \tag{4.50}$$

The solution of Eq. (4.50) is a valid descent direction for $f(\mathbf{x})$ since

$$\mathbf{h_{gn}}^T f'(\mathbf{x}) = \mathbf{h_{gn}}^T(\mathbf{J}^T F) = -\mathbf{h_{gn}}^T(\mathbf{J}^T\mathbf{J})\mathbf{h_{gn}} < 0 \tag{4.51}$$

The conventional Gaussian-Newton algorithm iterates along the descent direction h_{gn} with a constant step size until a minimum is reached. Based on the Gaussian-Newton algorithm Levenberg [1944] and Marquardt [1963] proposed the Levenberg-Marquardt algorithm by introducing a damped parameter μ to seek for a descent direction, which is the solution of Eq. (4.52)

$$(\mathbf{J}^T\mathbf{J} + \mu\mathbf{I})\mathbf{h}_{lm} = -\mathbf{J}^T\mathbf{F} \quad \mu \geq 0 \tag{4.52}$$

To deal with the constraints, a global and projected Levenberg-Marquardt method was proposed by Kanzow *et al.* [2005], which is a hybrid method of gradient projection and the unconstrained Levenberg-Marquardt method. The iterative formula of the projected Levenberg-Marquardt algorithm is given in Eq. (4.53)

$$\mathbf{x}_{k+1} = P_X(\mathbf{x}_k + \mathbf{d}_k) \tag{4.53}$$

with P_X being the projection operator and $\mathbf{d}_k$ being the solution of Eq. (4.52) in each iteration. The damped parameter μ is updated in each iteration. The projected algorithm is summarized in Alg. 1.

Data: $\mathbf{x}_0$ and $\mu > 0$
Result: the minimizer $\mathbf{x}^*$
set k=0, found=false;
while *found==false* **do**
  if $\mathbf{F}(\mathbf{x}) == 0$ **then**
    found=true, stop
  end
  compute $\mathbf{J}_k$;
  $\mu_k = \mu\|\mathbf{F}(\mathbf{x}_k)\|^2$;
  solve $(\mathbf{J_k}^T\mathbf{J_k} + \mu_k\mathbf{I})\mathbf{h}_{lm} = -\mathbf{J_k}^T\mathbf{F_k}$;
  $\mathbf{x}_k = P_X(\mathbf{x}_k + \mathbf{d}_k)$;
  k=k+1 ;
end

Algorithm 1: Projected Levenberg-Marquardt Algorithm

The projected Levenberg-Marquardt algorithm is globalized by introducing a projected gradient step described by Bertsekas [1976] whenever the projected Levenberg-Marquardt step cannot provide a sufficient decrease. The global version of projected Levenberg-Marquardt algorithm is

summarized in Alg. 2

Data: $\mathbf{x}_0$ and $\mu > 0$, $\beta, \sigma, \gamma \in (0, 1)$
Result: the minimizer $\mathbf{x}^*$
set k=0, found=false;
while *found==false* **do**
  if $\mathbf{F}(\mathbf{x}) == 0$ **then**
    found=true, stop;
  end
  compute $\mathbf{J}_k$;
  $\mu_k = \mu \|\mathbf{F}(\mathbf{x}_k)\|^2$;
  solve $(\mathbf{J_k}^T\mathbf{J_k} + \mu_k\mathbf{I})\mathbf{h}_{lm} = -\mathbf{J_k}^T\mathbf{F_k}$;
  if $\|\mathbf{F}(P_X(\mathbf{x}_k + \mathbf{d}_k))\| \leq \gamma\|\mathbf{x_k}\|$ **then**
    $\mathbf{x}_{k+1} = P_X(\mathbf{x}_k + \mathbf{d}_k)$;
    $k = k + 1$;
    continue ;
  else
    compute a step size $s_k = \max\{\beta^l | l = 0, 1, 2, ...\}$ such that
    $f(\mathbf{x}_k(s_k)) \leq f(\mathbf{x}_k) + \sigma\nabla f(\mathbf{x}_k)^T(\mathbf{x}_k(s_k) - \mathbf{x}_k)$ with $\mathbf{x}_k(s) = P_X(\mathbf{x}_k - s\nabla f(\mathbf{x}_k)$;
    $\mathbf{x}_{k+1} = \mathbf{x}_k(s_k)$;
    $k = k + 1$;
  end
end

Algorithm 2: Global Version of the Projected Levenberg-Marquardt Algorithm

4.1.6 Semi-Parametric and Non-Parametric Models

In this section, we briefly review Gaussian Mixture Models (GMM) and Kernel Density Estimation (KDE).

4.1.6.1 Gaussian Mixture Models

A Gaussian mixture model is a semi-parametric model, which assumes that the data distribution is a mixture of parametric models, for instance a sum of Gaussian distributions. A random variable X follows a Gaussian mixture distribution with weights α_i if its PDF can be written as

$$f(x|\Theta) = \sum_{i=1}^{M} \alpha_i f_i(X|\theta_i) \tag{4.54}$$

where $\Theta = (\alpha_1, \theta_1, \alpha_2, \theta_2, \ldots, \alpha_M, \theta_M)$ and $f_i(X|\theta_i)$ is a Gaussian distribution governed by the mean and covariance $\theta_i = (\mu_i, \Sigma_i)$. The standard method to estimate the parameters of a GMM is maximum likelihood estimation which is solved through the Expectation-Maximization (EM) algorithm by Dempster *et al.* [1977]. It is based on the assumption of missing variables y_i associated with incomplete samples $x_j, j = 1, ...N$. The assumed missing variables taking values from $[1, M]$ are the labels denoting which component produced the sample x_j. The introduction of missing variables can simplify the computation of the likelihood function to be optimized as the likelihood function includes the log of a sum. After introducing the hidden variables, the

likelihood function for the complete data $Z = (X, Y)$ is

$$\log[L(\Theta|X, Y)] = \sum_{j=1}^{N} \log[\alpha_{y_j} f_{y_j}(x_j|\theta_{y_j})] \tag{4.55}$$

The EM algorithm performs parameter estimation by iteratively applying two steps until convergence is reached. The first step is to compute the expectation of the complete likelihood while the second step is to maximize this function in order to find the best parameter values.

E-step: Compute the expectation of the complete likelihood function with respect to the missing variables given the data.

$$Q(\theta, \theta^{n-1}) = E[\log f(x, y|\theta)|x, \theta] \tag{4.56}$$

which results in the probability of the data from each component.

$$f(i|x_j, \Theta) = \frac{\alpha_i f_i(x_j|\theta_i)}{\sum_{k=1}^{M} \alpha_k f_k(x_j|\theta_k)} \tag{4.57}$$

M-step: Maximize the expectation to derive the involved parameters.

$$\theta^n = \underset{\theta}{\text{argmax}} Q(\theta, \theta^{n-1}) \tag{4.58}$$

The updating equations for the parameters are

$$\alpha_i^n = \frac{1}{N} \sum_{j=1}^{N} f(i|x_j, \Theta^{n-1}) \tag{4.59}$$

$$\mu_i^n = \frac{\sum_{j=1}^{N} x_j f(i|x_j, \Theta^{n-1})}{\sum_{j=1}^{N} f(i|x_j, \Theta^{n-1})} \tag{4.60}$$

$$\Sigma_i^n = \frac{\sum_{j=1}^{N} f(i|x_j, \Theta^{n-1})(x_j - \mu^i)(x_j - \mu^i)^T}{\sum_{j=1}^{N} f(i|x_j, \Theta^{n-1})} \tag{4.61}$$

The basic EM algorithm has some drawbacks, concerning mainly its initialization, i.e., the initial values of the parameters and the number of components. To overcome these drawbacks, the greedy EM algorithm of Vlassis & Likas [2002] and the Figueiredo-Jain algorithm by Figueiredo & Jain [2002] were proposed. We initialize the algorithm by k-means clustering and select the number of components by the Bayesian Information Criterion (BIC), defined as

$$BIC(M_k, X) = 2 \log L(X|\Theta_k, M_k) - d(M_k) \log N \tag{4.62}$$

where Θ_k is the set of parameters characterizing the model M_k, $d(M_k)$ is the number of parameters in Θ_k and N is the number of training samples.

4.1.6.2 Kernel Density Estimation (KDE)

Let $(x_1, x_2, ..., x_n)$ be n training samples drawn independently from an unknown PDF $p_X(x)$. Then the kernel density estimator $\hat{p}_X(x)$ of the unknown distribution is defined as

$$\hat{p}_X(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h}) \tag{4.63}$$

where h is the smoothing parameter (also called bandwidth), $K(.)$ is the kernel function which is a valid PDF. The estimator $\hat{p}_X(x)$ is a weighted mixture of a series of kernel functions being centered at each data point. Because the smoothing parameter is critical to the estimation quality, the method given in Shimazaki & Shinomoto [2010] for computing the best bandwidth is used in this thesis. The kernel function we applied is the Gaussian kernel shown in Eq. (4.64) with a standard deviation σ.

$$K(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{x^2}{2\sigma^2}) \tag{4.64}$$

4.1.7 Goodness-of-Fit Test

As we work with various statistical models, we need a quantitative measure to assess their goodness of fit. A goodness-of-fit measure evaluates the discrepancy between an empirical histogram and an estimated model; thus it computes the confidence that the data is generated from the assumed model. In this section, we describe three goodness-of-fit measures. The first one is the Kolmogorov-Smirnov (KS) distance, which uses the largest discrepancy between the empirical CDF and the CDF of the estimated model as a quantitative measure. The second one is the correlation coefficient, which determines the correlation between the two discrete vectors of probabilities. This is particularly helpful especially when no analytical CDF is available, which is the case of the GGR and GΓD. The third measure is the mean squared error, which is the sum of the squared errors between the histogram and the estimated PDF evaluated at the histogram bin centers.

4.1.7.1 Kolmogorov-Smirnov Distance

Given the empirical CDF $F_n(x)$ and the CDF $F(x)$ of the estimated model, the Kolmogorov-Smirnov distance explained by William *et al.* [2007] is defined as Eq. (4.65), and demonstrated in Fig. 4.2

$$D = \max_x |F_n(x) - F(x)| \tag{4.65}$$

After computing the Kolmogorov distance D, the significance level (p value) can be calculated based on the KS distribution using $Q_{ks}(d) = 1 - F_{ks}(d)$, where $F_{ks}(d)$ is the CDF of the KS distribution, defined in Eq. (4.66)

$$F_{ks}(d) = 1 - 2\sum_{i=1}^{\infty}(-1)^{i-1} \exp -2i^2d^2 \tag{4.66}$$

A small p value means that it is likely that the data is generated from the assumed distribution; otherwise, the hypothesis should be rejected. On the other hand, the KS distance can be used to compare different models. A model with a small KS distance is better.

4.1.7.2 Correlation Coefficient

The correlation coefficient is a simple measure of linear dependence between two datasets. It is defined as the covariance of the two variables X and Y divided by the product of their standard deviations, as shown in Eq. (4.67). It can be used as a linear similarity measure of two vectors. A small correlation means that two vectors are similar. If applied to a histogram, it can be used to compare different models. The model with the smaller correlation is better.

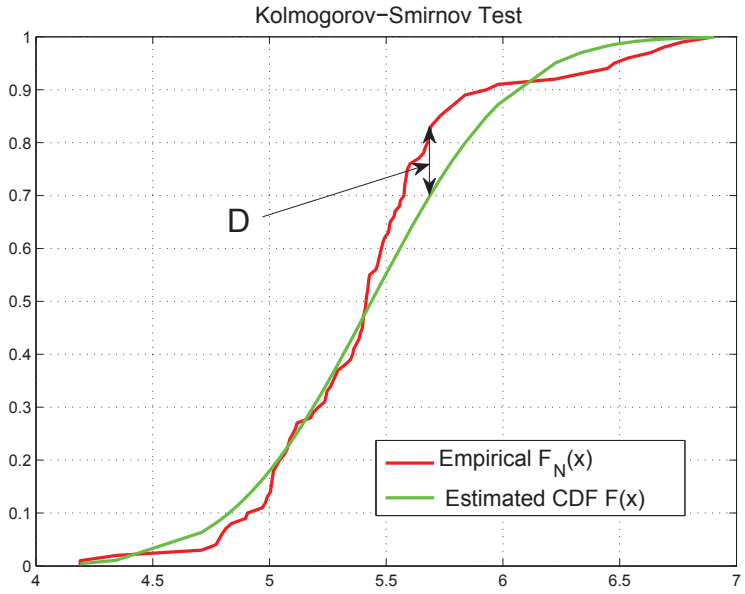


Figure 4.2: Kolmogorov-Smirnov statistic D.

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{4.67}$$

4.1.7.3 Mean Squared Error (MSE)

Similar to the correlation coefficient, MSE is also a measure of closeness of the estimated model to the histogram. It is defined as the average squared error between the histogram and the estimated PDF evaluated at the bin centers and shown in Eq. (4.68).

$$D = \frac{1}{N}\sum_{i=1}^{N} \|p_n(x_i) - p(x_i)\|^2 \tag{4.68}$$

with $p_n(x)$ is the histogram and $p(x)$ is the estimated PDF. Note that the summation is applied to the bin centers, not to the observed data. It can be seen that the correlation coefficient and the MSE depend on the number of bins because they compare the closeness of the estimated PDF to the histogram.

4.1.7.4 Evaluation of the Models

In this section, ten selected models, listed in Table 4.1 with their corresponding MoLC equations, are evaluated on 20 classes using the accuracy measures presented in the previous section. Example images of the 20 classes are shown in Fig. 5.5. The size of each test image is 160×160 pixels. Table 4.2 lists the number of images of each category in the first column together with the class index. The best accurate model for each class is highlighted in red. The CDFs of the non-analytical models are computed by numerical integration. The number of bins used for histogram

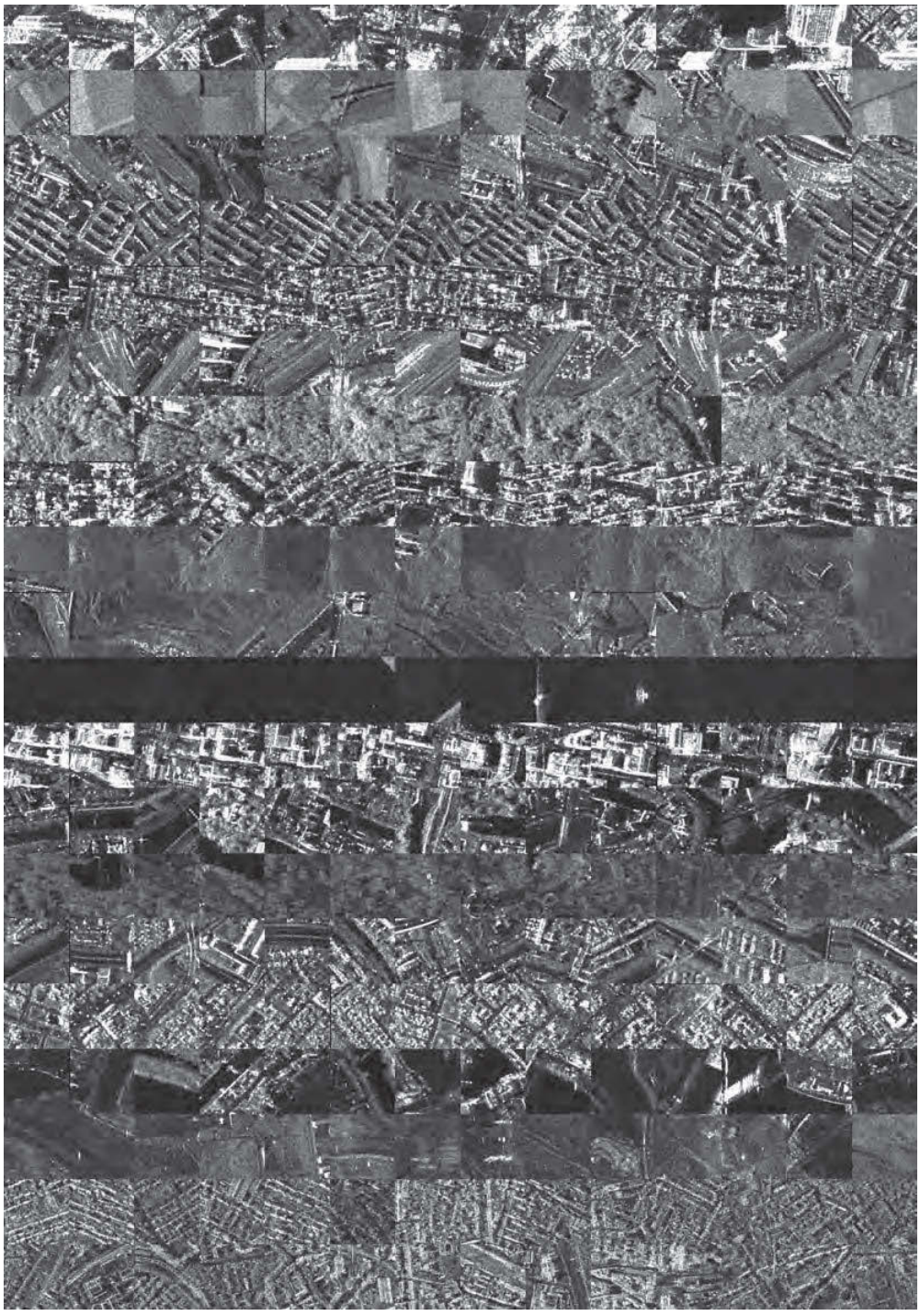


Figure 4.3: Example images of 20 classes for statistical model evaluation.

computation is 100. In this table, HTR, WBL, LGN, NKG are abbreviations of the Heavy Tailed Rayleigh, the Weibull, the log-normal, and the Nakagami distribution. Unfortunately the computation of the HTR CDF is quite computationally-intensive, therefore, the KS distances have

Table 4.1: The amplitude PDFs and MoLC equations of the ten models. $\Phi(i,x)$ is the i^{th} order polygamma function.

Model	PDF	MoLC equations
$\Gamma^{1/2}$	$p(x)=\frac{2}{\Gamma(L)}(\frac{L}{\mu})^L x^{2L-1}\exp\Big(-\frac{Lx^2}{\mu}\Big),\quad L,\mu>0$	$2\kappa_1=\Phi(0,L)+\ln\mu-\ln L$ $4\kappa_2=\Phi(1,L),$
HTR	$p(x)=x\int_0^{+\infty}\rho\exp(-\gamma\rho^\alpha)J_0(x\rho)d\rho,\quad \alpha,\gamma>0$	$\kappa_1=-\Phi(0,1)\frac{1-\alpha}{\alpha}+\ln 2\gamma^{1/\alpha}$ $\kappa_2=\frac{\phi(1,1)}{\alpha^2}$
GGR	$p(x)=\frac{\gamma^2 x}{\lambda^2\Gamma^2(\lambda)}\int_0^{\frac{\pi}{2}}\exp\Big(-(x\gamma)^{\frac{1}{\lambda}}(\lvert\cos\theta\rvert^{\frac{1}{\lambda}}+\lvert\sin\theta\rvert^{\frac{1}{\lambda}})\Big)d\theta,$ $\lambda,\gamma>0$	$\kappa_1=\lambda\Phi(0,2\lambda)-\ln\gamma-\lambda G_1(\lambda)G_0^{-1}(\lambda)$ $\kappa_2=\lambda^2[\Phi(1,2\lambda)+\frac{G_2(\lambda)}{G_0(\lambda)}-(\frac{G_1(\lambda)}{G_0(\lambda)})^2]$
GΓR	$p(x)=\frac{\nu}{2\eta\Gamma(\kappa)}\Big(\frac{\lvert x\rvert}{\eta}\Big)^{\kappa\nu-1}\exp\Big(\frac{\lvert x\rvert}{\eta}\Big)^{\nu},\quad \nu,\kappa,\eta>0$	$\kappa_1=\ln\eta+\varpi\Phi_0(2\kappa)-\varpi\frac{G_1(\kappa,\varpi)}{G_0(\kappa,\varpi)}$ $\kappa_2=\varpi\Big[\Phi_0(1,2\kappa)+\frac{G_2(\kappa,\varpi)}{G_0(\kappa,\varpi)}-\frac{G_1^2(\kappa,\varpi)}{G_0^2(\kappa,\varpi)}\Big]$ $\kappa_3=\varpi^3\Big[\Phi_0(2,2\kappa)-\frac{G_3(\kappa,\varpi)}{G_0(\kappa,\varpi)}+$ $3\frac{G_2(\kappa,\varpi)G_1(\kappa,\varpi)}{G_0^2(\kappa,\varpi)}-2\frac{G_1^3(\kappa,\varpi)}{G_0^3(\kappa,\varpi)}\Big],\quad \varpi=1/\nu$
$\mathcal{K}$-root	$p(x)=\frac{4(\lambda L)^{\frac{L+\alpha}{2}}}{\Gamma(L)\Gamma(\alpha)}x^{L+\alpha-1}K_{\alpha-L}(2x\sqrt{\lambda L}),$ $L,\alpha,\lambda>0$	$2\kappa_1=\Phi(\alpha)+\Phi(L)-\ln\lambda L$ $2^i\kappa_i=\Phi(i-1,\alpha)+\Phi(i-1,L)$ $i=2,3,\cdots$
$\mathcal{G}^0$	$p(x)=\frac{2L^L\Gamma(L-\alpha)}{\gamma^\alpha\Gamma(L)\Gamma(-\alpha)}\frac{x^{2L-1}}{(\gamma+Lx^2)^{L-\alpha}},\quad L,\gamma>0,\alpha<0$	$2\kappa_1=\ln\gamma/L+\Phi(L)-\Phi(-\alpha)$ $2^i\kappa_i=\Phi(i-1,L)+(-1)^i\Phi(i-1,-\alpha)$ $i=2,3,\cdots$
GΓD	$p(x)=\frac{\beta x^{\beta\lambda-1}}{\alpha^{\beta\lambda}\Gamma(\lambda)}\exp\Big(-(\frac{x}{\alpha})^\beta\Big),\quad \alpha,\beta,\lambda>0$	$\kappa_1=\ln\alpha+\frac{\Phi(0,\lambda)}{\beta}$ $\kappa_i=\frac{\Phi(i-1,\lambda)}{\beta^i}\quad i=2,3,\cdots$
LGN	$p(x)=\frac{1}{\sqrt{2\pi}\sigma x}\exp\Big[-\frac{(\ln x-m)^2}{2\sigma^2}\Big],\quad \sigma>0,m\in\mathbb{R}$	$\kappa_1=m$ $\kappa_2=\sigma^2$
WBL	$p(x)=\frac{\eta}{\mu^\eta}x^{\eta-1}\exp\Big[-(\frac{x}{\mu})^\eta\Big],\quad \mu,\eta>0$	$\kappa_1=\ln\mu+\eta^{-1}\Phi(0,1)$ $\kappa_2=\eta^{-2}\Phi(1,1)$
NKG	$p(x)=\frac{2}{\Gamma(L)}(\lambda L)^L x^{2L-1}\exp\Big[-\lambda Lx^2\Big],\quad L,\lambda>0$	$2\kappa_1=\Phi(0,L)-\ln\lambda-\ln L$ $4\kappa_2=\Phi(1,L)$

$G_v(\lambda)=\int_0^{\pi/2}\frac{\ln^v A(\theta,\lambda)}{A(\theta,\lambda)^{2\lambda}}d\theta,\quad A(\theta,\lambda)=\lvert\cos\theta\rvert^{1/\lambda}+\lvert\sin\theta\rvert^{1/\lambda}$

$G_i(\kappa,\varpi)=\int_0^{\pi/2}\lvert\cos\theta\sin\theta\rvert^{\frac{\kappa}{\varpi}-1}\frac{\ln^i A(\theta,\varpi)}{A^{2\kappa}(\theta,\varpi)}d\theta$

been omitted. From Table. 4.2, we can see that in all cases the correlation coefficient and the MSE behave similarly, while the KS distance shows a slightly different performance. The first observation is that there is no universal best model for all classes, as shown in Singh *et al.* [2012]. Thus, the model selection should be application orientated. Generally, $\mathcal{G}^0$ performs quite well for most classes. For fully developed texture images, like the classes $2,3,9,10$, GΓR performs well while $\mathcal{G}^0$ has a high accuracy especially for most urban areas. Surprisingly, for some classes, the log-normal distribution does not perform badly. Out of all the models, GGR and HTR never perform best for all classes. The reason is probably the numerical integral, which does not only increase the computational burden, but also decreases the accuracy.

Table 4.2: Evaluation of statistical models.

Class	Measure	$\Gamma^{1/2}$	GGR	GΓR	HTR	$\mathcal{K}$-root	$\mathcal{G}^0$	GΓD	LGN.	WBL	NKG.
1-314	CC	0.8900	0.9418	0.9555	0.9420	0.7306	0.9716	0.9392	0.9695	0.9118	0.8900
	MSE(1e7)	0.8354	0.5473	0.3602	0.5176	2.4385	0.2144	0.5392	0.3004	0.7228	0.8354
	KS	0.1255	0.0884	0.0738	–	0.2022	0.0453	0.4627	0.0561	0.1121	0.1255
2-369	CC	0.9468	0.7987	0.9526	0.6494	0.9603	0.9540	0.9486	0.9374	0.9375	0.9468
	MSE(1e6)	0.9828	3.9736	0.9015	5.3704	0.7453	0.8604	0.9041	1.0631	1.2169	0.9828
	KS	0.0557	0.1732	0.0414	–	0.0415	0.0464	0.5976	0.0584	0.0664	0.0557
3-320	CC	0.9495	0.9402	0.9718	0.9221	0.9332	0.9691	0.9618	0.9673	0.9510	0.9495
	MSE(1e6)	1.7427	1.9563	1.7393	2.3528	1.7876	1.9123	1.8324	1.7490	1.7606	1.7427
	KS	0.0841	0.0953	0.0512	–	0.0902	0.0436	0.6049	0.0569	0.0842	0.0841

Continued on next page

Table 4.2 – continued from previous page

Class	Measure	$\Gamma^{1/2}$	GGR	GΓR	HTR	$\mathcal{K}$-root	$\mathcal{G}^0$	GΓD	LGN.	WBL	NKG.
4-172	CC	0.9170	0.9423	0.9706	0.9348	0.8657	0.9611	0.9617	0.9729	0.9261	0.9170
	MSE(1e6)	0.2513	0.1860	0.0943	0.2110	0.4836	0.1285	0.1209	0.0904	0.2279	0.2513
	KS	0.0910	0.0715	0.0407	–	0.1300	0.0450	0.9149	0.0369	0.0847	0.0910
5-194	CC	0.9007	0.9409	0.9688	0.9365	0.7957	0.9722	0.9510	0.9764	0.9158	0.9007
	MSE(1e6)	0.0438	0.0288	0.0154	0.0302	0.1165	0.0138	0.0236	0.0125	0.0383	0.0438
	KS	0.1126	0.0831	0.0534	–	0.1724	0.0410	0.6211	0.0416	0.1028	0.1126
6-104	CC	0.9526	0.9462	0.9832	0.9403	0.9287	0.9817	0.9696	0.9813	0.9538	0.9526
	MSE(1e6)	0.0953	0.1474	0.0449	0.1790	0.1487	0.0486	0.0648	0.0500	0.1014	0.0953
	KS	0.0864	0.0914	0.0452	–	0.1006	0.0334	0.5448	0.0482	0.0859	0.0864
7-198	CC	0.9674	0.9533	0.9674	0.9349	0.9615	0.9644	0.9675	0.9375	0.9668	0.9674
	MSE(1e6)	0.2863	0.4253	0.2792	0.5742	0.3140	0.3033	0.2705	0.5097	0.2942	0.2863
	KS	0.0307	0.0492	0.0287	–	0.0343	0.0315	0.4016	0.0630	0.0326	0.0307
8-211	CC	0.8637	0.9144	0.9409	0.9059	0.7183	0.9478	0.9207	0.9531	0.8832	0.8637
	MSE(1e6)	0.0755	0.0528	0.0357	0.0554	0.1914	0.0306	0.0487	0.0307	0.0670	0.0755
	KS	0.1249	0.0936	0.0750	–	0.2038	0.0595	0.5304	0.0591	0.1143	0.1249
9-80	CC	0.9485	0.8017	0.9607	0.6413	0.9604	0.9555	0.9498	0.9591	0.9273	0.9485
	MSE(1e6)	1.8130	6.6407	1.4947	10.1411	1.5479	1.8191	1.6209	1.5321	2.5065	1.8130
	KS	0.0619	0.1889	0.0403	–	0.0408	0.0341	0.6814	0.0413	0.0842	0.0619
10-50	CC	0.9665	0.8928	0.9787	0.7970	0.9757	0.9780	0.9714	0.9770	0.9551	0.9665
	MSE(1e6)	0.4842	1.5594	0.3946	2.9038	0.3954	0.4189	0.6701	0.4184	0.6499	0.4842
	KS	0.0676	0.1433	0.0419	–	0.0461	0.0366	0.7351	0.0417	0.0806	0.0676
11-210	CC	0.6172	0.5846	0.6260	0.5222	0.8098	0.8003	0.6101	0.6140	0.6143	0.6172
	MSE(1e3)	0.9696	0.9924	0.9692	1.0446	0.2864	0.2687	0.9743	0.9795	0.9717	0.9696
	KS	0.0476	0.1161	0.0372	–	0.0551	0.0411	0.3095	0.0696	0.0596	0.0476
12-204	CC	0.9192	0.9661	0.9824	0.9589	0.8079	0.9745	0.9719	0.9839	0.9402	0.9192
	MSE(1e7)	0.6375	0.3153	0.1589	0.3437	1.8698	0.1970	0.2624	0.1468	0.5012	0.6375
	KS	0.1046	0.0617	0.0400	–	0.1692	0.0416	0.8576	0.0346	0.0888	0.1046
13-109	CC	0.8942	0.9260	0.9396	0.9281	0.7796	0.9655	0.9289	0.9525	0.9083	0.8942
	MSE(1e4)	0.1781	0.1805	0.1830	0.1890	0.0085	0.0023	0.1854	0.1783	0.1781	0.1781
	KS	0.1122	0.0890	0.0707	–	0.1677	0.0425	0.4257	0.0570	0.1034	0.1122
14-114	CC	0.9437	0.9081	0.9514	0.8367	0.9447	0.9468	0.9490	0.9389	0.9378	0.9437
	MSE(1e5)	0.1213	0.1828	0.1024	0.3019	0.1221	0.1106	0.1084	0.1181	0.1328	0.1213
	KS	0.0482	0.0939	0.0341	–	0.0412	0.0362	0.6862	0.0527	0.0570	0.0482
15-67	CC	0.9440	0.9548	0.9793	0.9528	0.9015	0.9770	0.9675	0.9769	0.9490	0.9440
	MSE(1e6)	0.1527	0.1602	0.0765	0.2092	0.2950	0.0686	0.1044	0.1105	0.1454	0.1527
	KS	0.0859	0.0781	0.0430	–	0.1176	0.0353	0.5881	0.0457	0.0821	0.0859
16-158	CC	0.9289	0.9488	0.9813	0.9513	0.8634	0.9869	0.9636	0.9853	0.9361	0.9289
	MSE(1e6)	0.2415	0.1944	0.0684	0.1873	0.5568	0.0498	0.1304	0.0601	0.2242	0.2415
	KS	0.0972	0.0807	0.0422	–	0.1396	0.0264	0.5474	0.0343	0.0918	0.0972
17-110	CC	0.8449	0.8727	0.9068	0.8638	0.7599	0.9267	0.8940	0.9253	0.8576	0.8449
	MSE(1e6)	1.8957	1.6404	1.2114	1.8171	2.5503	1.5552	0.8923	1.1517	1.7874	1.8957
	KS	0.1365	0.1182	0.0951	–	0.1806	0.0616	0.5198	0.0798	0.1295	0.1365
18-124	CC	0.9164	0.8327	0.9327	0.7268	0.9282	0.9281	0.9264	0.9350	0.9010	0.9164
	MSE(1e5)	1.4271	2.1532	1.3530	3.0445	1.3899	1.6351	1.3977	1.3372	1.5571	1.4271
	KS	0.0660	0.1485	0.0412	–	0.0483	0.0292	0.7040	0.0389	0.0828	0.0660
19-279	CC	0.9506	0.9539	0.9923	0.9557	0.9410	0.9946	0.9805	0.9922	0.9530	0.9506
	MSE(1e5)	0.1013	0.1162	0.0181	0.1156	0.1454	0.0133	0.0409	0.0188	0.1018	0.1013
	KS	0.0890	0.0859	0.0309	–	0.0982	0.0147	0.7642	0.0312	0.0874	0.0890
20-195	CC	0.9436	0.9572	0.9894	0.9588	0.9118	0.9914	0.9753	0.9903	0.9497	0.9436
	MSE(1e5)	0.8742	0.8915	0.2025	0.8696	1.6198	0.1286	0.4624	0.1821	0.8464	0.8742
	KS	0.0943	0.0800	0.0353	–	0.1177	0.0197	0.6722	0.0328	0.0892	0.0943

4.2 Information Similarity Metrics

4.2.1 Shannon Entropy

Definition 1. The *entropy* of a discrete random variable X with a probability mess function $p_X(x) = Pr(X = x)$ is defined as

$$H(X) = -\sum_{x\in\mathcal{X}} p_X(x)\log p_X(x)^{1} \tag{4.69}$$

Statistically, entropy is the expectation of $-\log\frac{1}{p_X(x)}$. It can be proven that the entropy of a discrete random variable is always positive because $0 \leq p_X(x) \leq 1$. Furthermore, the entropy is always invariant to an invertible transformation. In the context of image analysis, the entropy can be considered as a measure of uncertainty captured by the dispersion of the probability distribution of image intensities. Images with a uniform intensity distribution have a high dispersion and therefore a higher entropy. On the contrary, images with intensities of high variability have lower entropy, which is the basic idea of the Kadir-Brady saliency detector of Kadir & Brady [2001].

In the case of a continuous random variable, the summation should be replaced by an integral, thus called differential entropy. In contrast to the discrete case, the differential entropy of a continuous random variable can be negative because $p_X(x)$ can take values outside of the interval $[0, 1]$. In addition, the differential entropy is not invariant to invertible transformations. Given a continuous random variable $X \sim p_X(x)$ and an invertible transformation $Y = f(X)$ and $X = g(Y)$, the PDF of Y is given by

$$p_Y(y) = \frac{p_X(x)}{|f'(x)|} = p_X(g(y))|g'(y)| \tag{4.70}$$

Therefore, the entropy of Y is

$$H(Y) = -\int p_Y(y)\log p_Y(y)dy = H(X) + E(\log|f'(x)|) \tag{4.71}$$

Generally, the entropy is not invariant to invertible transformations. The concept of entropy can be generalized to a random vector, thus leading to joint entropy.

Definition 2. The *joint entropy* of two random variables X and Y, with their joint PDF $p_{X,Y}(x, y)$ is given by

$$H(X, Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{X,Y}(x, y)\log p_{X,Y}(x, y) \tag{4.72}$$

Similarly, the joint entropy of a discrete random vector is invariant to invertible transformations. In the case of continuous variables, the relation of Eq. (4.71) still holds when we replace the differential by a Jacobian. In the special case of a linear transformation, the following relations hold.

$$H(aX) = H(X) + \log|a| \tag{4.73}$$

$$H(AX) = H(X) + \log|\det A| \tag{4.74}$$

[1]For continues random variables, the summation in the definition of the information similarity measures has to be replaced with an integral.

where a is a scalar and A is a matrix. In principle, if two images are independent, their joint entropy is equal to the sum of their individual entropies. On the other hand, the more similar two images are, the lower their joint entropy is compared to the sum of the individual entropies Tourassi *et al.* [2007].

Definition 3. *Conditional entropy* is a quantity measuring how much information is remaining when we regard one random variable through another, it is defined as

$$H(X|Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{X,Y}(x,y)\log p_{X|Y}(x|y) \tag{4.75}$$

where $p_{X|Y}(x|y)$ is the conditional distribution. Similar to joint entropy, the conditional entropy of an image, given another known image, is low if the two images are correlated.

Based on the chain rule, it can be proven that these three kinds of entropies have the following relation

$$H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y) \tag{4.76}$$

4.2.2 Kullback-Leibler Divergence

While the entropy describes the uncertainty of a random variable, the Kullback-Leibler divergence (or relative entropy) is a distance measure between two distributions.

Definition 4. The *Kullback-Leibler divergence* between two distributions $X \sim p_X(x)$ and $Y \sim p_Y(x)$ is defined as

$$D(X||Y) = \sum_x p_X(x)\log\frac{p_X(x)}{p_Y(x)} \tag{4.77}$$

If the two probability density functions are close to each other, their Kullback-Leibler divergence is small. In contrast, it is larger if there is a great deviation between the two density functions. It can be proven that the Kullback-Leibler divergence is always positive and becomes zero if and only if $p_X(x)$ and $p_Y(x)$ are identical. For continuous random variables, the summation has to be replaced by an integral. However, it can be noted that the Kullback-Leibler divergence is not a real distance metric since it is asymmetric. Therefore, a symmetrized version $D(X,Y) = D(X||Y) + D(Y||X)$, called Jeffreys information, is usually used as a similarity measure. Unlike entropy, a nice property of the Kullback-Leibler divergence is that it is invariant under transformations.

In most cases, the Kullback-Leibler divergence does not have a closed form expression, thus we have to rely on a numerical approximation Hershey & Olsen [2007] or Monte Carlo simulation Goldberger *et al.* [2003]. Fortunately, for most parametric distributions, like Gamma, Weibull, Log-Normal, GΓD, or GGD, analytical formulas are available, which can significantly speed up the computation. In principle, the Kullback-Leibler divergence can be generalized to multi-dimensional cases; however, due to the difficulty in modeling the joint density distribution, a higher order Kullback-Leibler divergence is usually not realistic for practical applications.

4.2.3 Mutual, Variational and Mixed Information

Definition 5. Given two random variables X and Y with a joint PDF $p_{X,Y}(x,y)$ and their marginal probability mess functions $p_X(x)$ and $p_Y(y)$, their *mutual information* is defined as

the Kullback-Leibler divergence between the joint distribution $p_{X,Y}(x,y)$ and the product of the marginal distributions $p_X(x)p_Y(y)$

$$I(X,Y) = D(p_{X,Y}(x,y)||p_X(x)p_Y(y)) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{X,Y}(x,y)\log\frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)} \tag{4.78}$$

Mutual information is always positive and is zero if and only if the two random variables X and Y are independent, which means that $p_{X,Y}(x,y) = p_X(x)p_Y(y)$. Generally, mutual information is invariant under any smooth and uniquely invertible transformation of the variables Kraskov *et al.* [2004]. Compared with the correlation coefficient, mutual information is a general measure to quantify the statistical dependence between two images by measuring the distance from the joint probability density to independence defined by the product of marginal densities, which is quite useful for the analysis of heterogeneous data. If the two images are independent, their mutual information decreases to zero. On the contrary, if the two images are similar, their mutual information is higher. By replacing the joint distribution with the product of the marginal distributions, one obtains the lautum [1] information Palomar & Verdu [2008] defined as Eq. (4.79).

$$L(X,Y) = D(p_X(x)p_Y(y)||p_{X,Y}(x,y)) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_X(x)P_Y(y)\log\frac{p_X(x)p_Y(y)}{p_{X,Y}(x,y)} \tag{4.79}$$

Definition 6. *Variational information*, defined by Eq. (4.80), was proposed in Meila [2003] to compare clusters by measuring the amount of information lost and gained when changing from one cluster to another. It is defined by

$$V(X,Y) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{X,Y}(x,y)\log\frac{p_{X,Y}(x,y)^2}{p_X(x)p_y(y)}dxdy \tag{4.80}$$

While mutual information quantifies the common information between X and Y, variational information quantifies the different information transmitted through X and Y. The relationship between these information measures is shown by a Venn diagram in Fig. 4.4, where $H(X)$ is the entropy and $H(X|Y)$ is the conditional entropy.

Based on mutual and variational information, mixed information was initially proposed in Gueguen & Datcu [2009] to unify mutual and variational information by introducing a parameter α to trade off between the common information and the different information of two random variables.

Definition 7. Mixed information is defined as

$$I_\alpha(X,Y) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{X,Y}(x,y)\log\frac{p_{X,Y}(x,y)^{1+\alpha}}{p_X(x)p_y(y)}dxdy \tag{4.81}$$

As α may vary between 0 and 1, I_α can be viewed as a mixture of mutual and variational information measures. In particular, when $\alpha = I(X,Y)/H(X,Y)$, the mixed information becomes zero $I_\alpha = 0$. In Gueguen *et al.* [2011b], a theoretical analysis of local mixed information for change detection is performed by dependence modeling, in addition to an approach to calculate the trade-off parameter.

[1] Lautum is the reverse spelling of mutual.

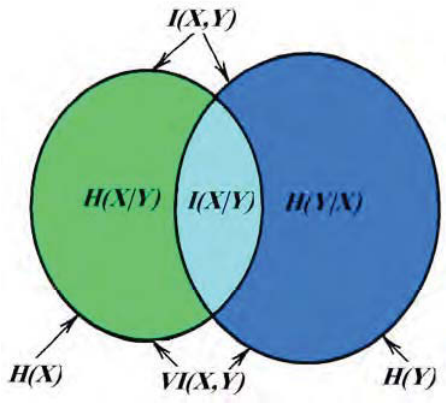


Figure 4.4: Venn diagram of information measures

4.3 A Benchmark for SAR Change Detection Evaluation

In this section, we focus on the simulation of SAR image statistics for the evaluation of change detection. In order to gain insight into different changes, seven kinds of changes are simulated using a patch from a TerraSAR-X image shown in Fig. 4.9(a), including random intensity changes, random permutations, first and second order statistical changes, texture changes and changes due to linear and nonlinear transformations.

4.3.1 Intensity Change Simulation

Intensity changes are achieved by replacing randomly selected pixel values of an image by samples drawn independently from a uniform distribution defined on (x_{min}, x_{max}), where x_{min} and x_{max} are the minimum and maximum pixel brightness values of the image. As an alternative, random permutation of the pixels is also included as it preserves the first order statistics (the histogram is exactly the same). Simulated images for these two cases are shown in Fig. 4.9(b) and (c). The changes are quite obvious and can be detected with high accuracy by most change detection methods.

4.3.2 First and Second Order Change Simulation

First order statistical changes are simulated by a probabilistic transformation of the pixel values. Recall that if a random variable X has a CDF $F_X(x)$, the transformed random variable $U = F_X(X)$ will follow a uniform distribution on the interval $[0, 1]$. By applying the inverse CDF of any distribution $Y \sim p_Y(y)$ to the uniform random variable U, we can get a desired random variable whose PDF is exactly $p_Y(y)$. In our case, we assume that the input image to be used for change simulation follows a $\mathcal{G}^0$ distribution (or any other valid distribution). This model has been demonstrated for high resolution SAR images by Frery *et al.* [1997]. Any reasonable model can be assumed for simulation, but the CDF of the assumed model has to be available. We transform the image pixels using the CDF of the $\mathcal{G}^0$ model (given in section 4.1.2) and then apply the inverse Gaussian CDF. We keep the mean and change the standard deviation σ to 0.6σ. The resulting pixels follow a Gaussian distribution exactly with the same mean. This is demonstrated in Fig. 4.5 using the image patch marked in Fig. 4.9(a). The simulation result is shown in Fig. 4.9(b). The algorithm is summarized in Alg. 3.

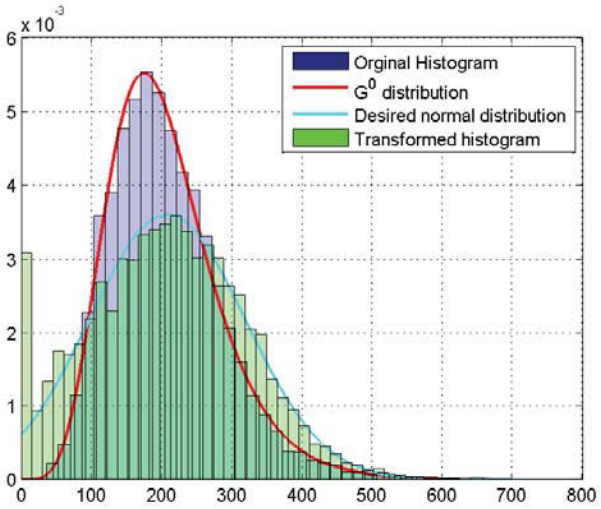


Figure 4.5: Change simulation in first order statistics using the patch shown in Fig. 4.9(a). The blue bars represent the histogram of the patch before transformation, which is assumed to follow a $\mathcal{G}^0$ distribution corresponding to the red curve. After transforming the pixels to a normal distribution, their histogram is computed (based on the green bars) using a normal distribution (cyan curve). It is worth noting that the mean value is preserved, but the variance is changed.

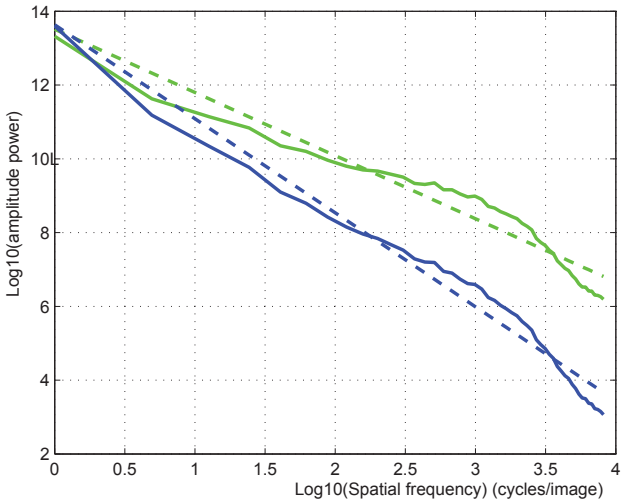


Figure 4.6: Changes in second order statistics by modifying the slope of the power spectrum of the test patch. The solid blue curve shows the amplitude power as a function of frequency on a log-log scale. The dashed blue line is the least squares straight line fit of the solid blue curve. Its slope is -1.7121. By multiplying the amplitude of each frequency by $f^{-\Delta\alpha}$ with $\Delta\alpha = 0.8$, the slope of the power spectrum changes to -2.5481, which corresponds to the dashed green line. The solid green curve is the plot of the amplitude power after multiplication as a function of frequency on a log-log scale.

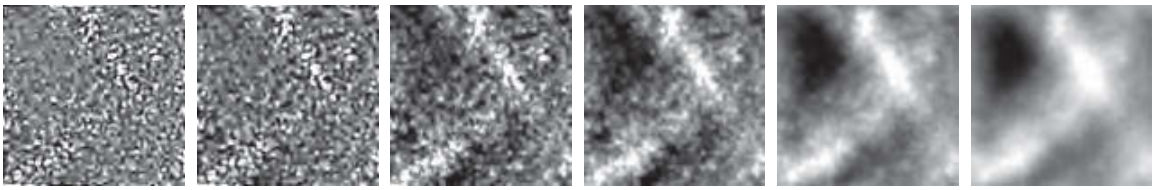
Data: The image I_{in} to be used for simulation.
Result: The simulated image I_{first}.
Estimate the CDF $F(x)$ of the image I_{in} ;
Apply $F(x)$ to obtain the uniform image I_{uni} ;
Compute the mean μ and standard deviation σ of I_{in} ;
Keep the mean $\mu' = \mu$ and change the standard deviation $\sigma' = 0.6\sigma$;
Determine G and G^{-1} ;
Apply the inverse CDF of the Gaussian distribution $G^{-1}(x)$ with mean μ' and std. σ' to the transformed image I_{uni} ;

Algorithm 3: First order statistical change simulation

Although the first order statistics provide considerable information about the probability distribution of the pixels, it cannot describe the spatial relationships of the image. Therefore, we go a step further to simulate changes in second order statistics, which considers the relation between pairs of pixels. The relation between pairs of pixels can be characterized by their auto-correlation function. Based on the Wiener-Khintchine theorem, stating that the auto-correlation function and the power spectrum form a Fourier transform pair, changes in second order statistics can be simulated by modifying the slope of the power spectrum. The power spectrum of an $N \times N$ image is defined as

$$S(u,v) = \frac{|\mathcal{F}(u,v)|^2}{N^2} \tag{4.82}$$

where $\mathcal{F}(u,v)$ is the Fourier transform of the image. By transforming the frequency $u = f\cos\theta$ and $v = f\sin\theta$ to polar coordinates and averaging over the orientations, it turns out that the power spectrum of a SAR image as a function of frequency on a log-log scale lies approximately on a straight line, as shown for our test patch by the dashed blue line in Fig. 4.6(a). In this case, the slope of the straight line is -1.7121. It is a second order statistic that can be modified by simulation. Any desired slope $\alpha + \Delta\alpha$ can be obtained by multiplying the amplitude power of each frequency by $f^{-\Delta\alpha}$. As an example, we chose $\Delta\alpha = 0.8$, which results in a simulated slope of -2.5481 and the image shown in Fig. 4.9(d). In addition, to characterize the influence of the slope, we used a series of $\Delta\alpha = (-1.2, -0.8, -0.2, 0.2, 0.8, 1.2)$ for simulation. The simulated results are shown in Fig. 4.7. As we increase $\Delta\alpha$, the image becomes smoother. Finally, we selected $\Delta\alpha = 0.8$ for change simulation. As can be seen from Fig. 4.9(d), it is not easy to visually discriminate the changes in second order statistics. The algorithm is summarized in Alg. 4.



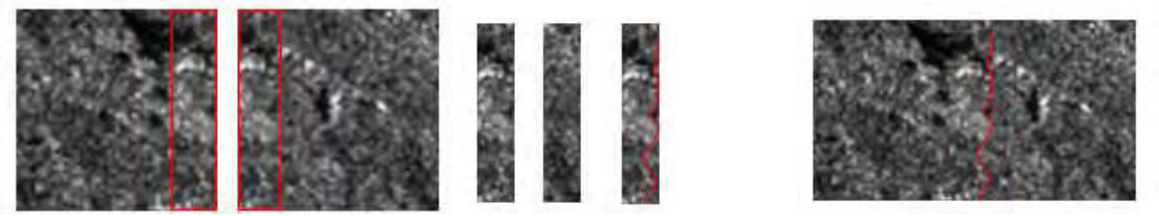
(a) (b) (c) (d) (e) (f)

Figure 4.7: Effect of changing $\Delta\alpha = (-1.2, -0.8, -0.2, 0.2, 0.8, 1.2)$ (from left to right) in second order change simulation. It is evident that with a negative $\Delta\alpha$ value, the high frequencies will be emphasized. On the contrary, with a positive $\Delta\alpha$ value, the images will be smoothed.

Data: The image I_{in} for simulation.
Result: The simulated image I_{second}.
Compute the normalized amplitude spectrum $S(u, v)$ and the phase spectrum $\phi(u, v)$;
Average the amplitude spectrum over all orientations ;
Compute the slope α by fitting a straight line ;
Change the power slope $\alpha' = \alpha + \Delta\alpha$;
Change the amplitude spectrum $A(u, v) = |F(u, v)|^2 \times (u^2 + v^2)^{\frac{\alpha'}{2}}$;
Keep the phase spectrum $\phi(u, v)$ and apply an inverse Fourier transformation using the new amplitude spectrum $A(u, v)$;

Algorithm 4: Second order statistical change simulation

4.3.3 Texture Change Simulation



(a) (b) (c)

Figure 4.8: Texture synthesis by image quilting for texture change simulation. (a) Two similar intersecting patches are selected. Their patch size is 100×100 pixels with an overlap of 20×20 pixels. If we stitch together the two patches directly, we would get an obvious discontinuity at the border. To reduce this effect, the error surface is computed within the overlapping part, which is the squared difference between two corresponding pixels. Dynamic programming is applied to find the minimum error bound (b), where the two patches match best (c).

To further evaluate the capabilities of change detection methods in detecting higher order changes, we simulated texture changes through texture synthesis. In the literature, Markov random fields were successfully used for texture synthesis by Paget & Longstaff [1998]. Similar texture patterns can be generated by re-using the parameters learned from the input images. However, due to the computational complexity, we chose the approach of image quilting by Efros & Freeman [2001] for texture synthesis because it is nonparametric and has low computational complexity. The main idea of this algorithm is to stitch together texture patches taken from the input image and it allows overlapping neighbor patches which are selected based on their similarity in the overlapping part. The novelty of this algorithm is that it allows neighboring patches to match smoothly at their boundary contours along curved lines. The algorithm achieves texture synthesis by iterating two steps. The first step is to choose randomly a small patch from the input image and then select another patch with high similarity within the overlapping part from all other potential patches. The second step is to calculate the error surface and determine the minimum error boundary between overlapping patches such that they match smoothly at the boundary. This is demonstrated in Fig. 4.8. The resulting image after simulation has a similar appearance as the input image, it is hard to discriminate with the naked eye; however,

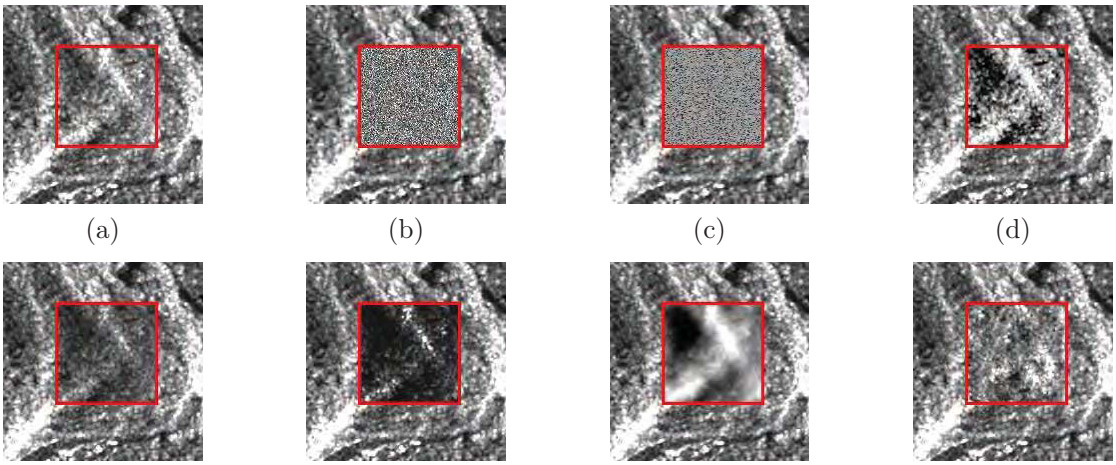
the images do differ in high order statistics.

Data: The image I_{in} for simulation.
Result: The simulated image $I_{texture}$.
Randomly select the first patch ;
while *the full image has not yet been covered.* **do**
  Select the most similar patch from the image I_{in} ;
  Compute the minimum boundary. ;
  Stich the two patches at the boundary ;
end

Algorithm 5: Texture change simulation

4.3.4 Linear and Nonlinear Change Simulation

Changes due to linear and nonlinear transformations can be easily achieved through linear and nonlinear transformations of the pixel values. Linear transformations are performed through $y = a(x - x_{min}) + x_{min}$. Nonlinear transformations are carried out through a nonlinear function, like $y = ax^2 + bx + c$ in our example. To test the robustness of the information similarity measures to noise, Gaussian white noise ($\sigma = 4$) is added to the synthesized images of each kind of simulation. All simulated images are shown in Fig. 4.9. The reason why we chose a quadratic function is that any nonlinear function can be approximated locally by a quadratic function.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 4.9: Simulated images: (a) The part selected by the red rectangle is used for simulation; (b) Random permutation; (c) Intensity change; (d) First order statistical change; (e) Linear change; (f) Nonlinear change; (g) Second order statistical change; (h) Textural change.

4.3.5 SAR Change Detection Based on Information Similarity Metrics

In general, change detection based on information similarity measures is performed locally in a sliding window by assessing the similarity of local patches from the two images. However, one critical issue is that we need an analytical expression of the information similarity measures. Otherwise, approximations Goldberger *et al.* [2003] or Monte Carlo simulations Hershey & Olsen [2007] have to be applied. See also the work by Inglada & Mercier [2007], where an Edgeworth

series was used to approximate the local PDFs. Monte Carlo simulation depends on sampling the estimated distribution to obtain a large amount of training samples. Although it can approximate the information measures precisely as long as the number of samples is sufficient, it is too time consuming to be applied to change detection with a sliding window. Unfortunately, for most distributions, there are almost no closed-form expressions for most information similarity measures. Therefore, we resort to the local average of the information measures as used in Gueguen *et al.* [2011b], except for the Kullback-Leibler divergence.

Assume $X(i,j)$ and $Y(i,j)$, $1 < i < M, 1 < j < N$ are two images, and a sliding window is applied to these two images for local information computation. Then the marginal and the joint probability density function of the pixels x_i and $y_i, i = 1, 2..., n$ in the sliding window from the two images are estimated. With these PDFs, the mutual information, the variational information, and the mixed information as change indicators can be estimated respectively as follows.

$$I(w_x, w_y) = \sum_{i=1}^{n} \log \frac{f(x_i, y_i)}{f(x_i) f(y_i)} \tag{4.83}$$

$$V(w_x, w_y) = -\sum_{i=1}^{n} \log \frac{f(x_i, y_i)^2}{f(x_i) f(y_i)} \tag{4.84}$$

$$I_\alpha(w_x, w_y) = \sum_{i=1}^{n} \log \frac{f(x_i, y_i)^\alpha}{f(x_i) f(y_i)} \tag{4.85}$$

As proposed by Gueguen etc. in Gueguen *et al.* [2011b], the α parameter in the mixed information is selected through a brute force search such that the accuracy is optimal.

$$\begin{aligned} KL_{\mathcal{G}^0}(f_1(x)||f_2(x)) = \ln \frac{B(n_2, -\alpha_2)}{B(n_1, -\alpha_1)} + n_1 \ln \frac{n_1}{\gamma_1} + n_2 \ln \frac{n_2}{\gamma_2} + (n_1 - n_2)[\ln \frac{\gamma_1}{n_1} + \Psi(n_1) \\ + \Psi(-\alpha_1)] - (n_1 - \alpha_1)[\Psi(n_1 - \alpha_1) - \Psi(-\alpha_1)] \\ + \frac{n_2 - \alpha_2}{B(n_1, -\alpha_1)} \int_0^\infty \frac{x^{n_1 - 1}}{(1+x)^{n_1 - \alpha_1}} \ln(1 + \frac{n_2 \gamma_1}{n_1 \gamma_2} x) dx \end{aligned} \tag{4.86}$$

$$\begin{aligned} KLD_{\mathrm{G\Gamma D}}\Big(f(x; \alpha_1, \beta_1, \lambda_1)||f(x; \alpha_2, \beta_2, \lambda_2)\Big) = -\lambda_1 - \lambda_2 + \Big(\frac{\alpha_1}{\alpha_2}\Big)^{\beta_2} \frac{\Gamma(\lambda_1 + \beta_2/\beta_1)}{\Gamma\lambda_1} \\ + \Big(\frac{\alpha_2}{\alpha_1}\Big)^{\beta_1} \frac{\Gamma(\lambda_2 + \beta_1/\beta_2)}{\Gamma\lambda_2} + (\beta_1\lambda_1 - \beta_2\lambda_2)\Big(\log \frac{\alpha_1}{\alpha 2} + \frac{\Psi(0, \lambda_1)}{\beta_1} - \frac{\Psi(0, \lambda_2)}{\beta_2}\Big) \end{aligned} \tag{4.87}$$

In the case of Kullback-Leibler divergence, there exist closed-form expressions for the parametric models presented in section 4.1.2 due to their good mathematical properties. The closed-form expressions of Kullback-Leibler divergence for the parametric $\mathcal{G}^0$ model and GΓD are given in Eq. (4.86) and Eq. (4.87) respectively. They are abbreviated as KLD_$\mathcal{G}^0$D, and KLD_GΓD in the following section. The integral in the last term of KLD_$\mathcal{G}^0$D can be computed using a Gauss-Kronrod quadrature algorithm.

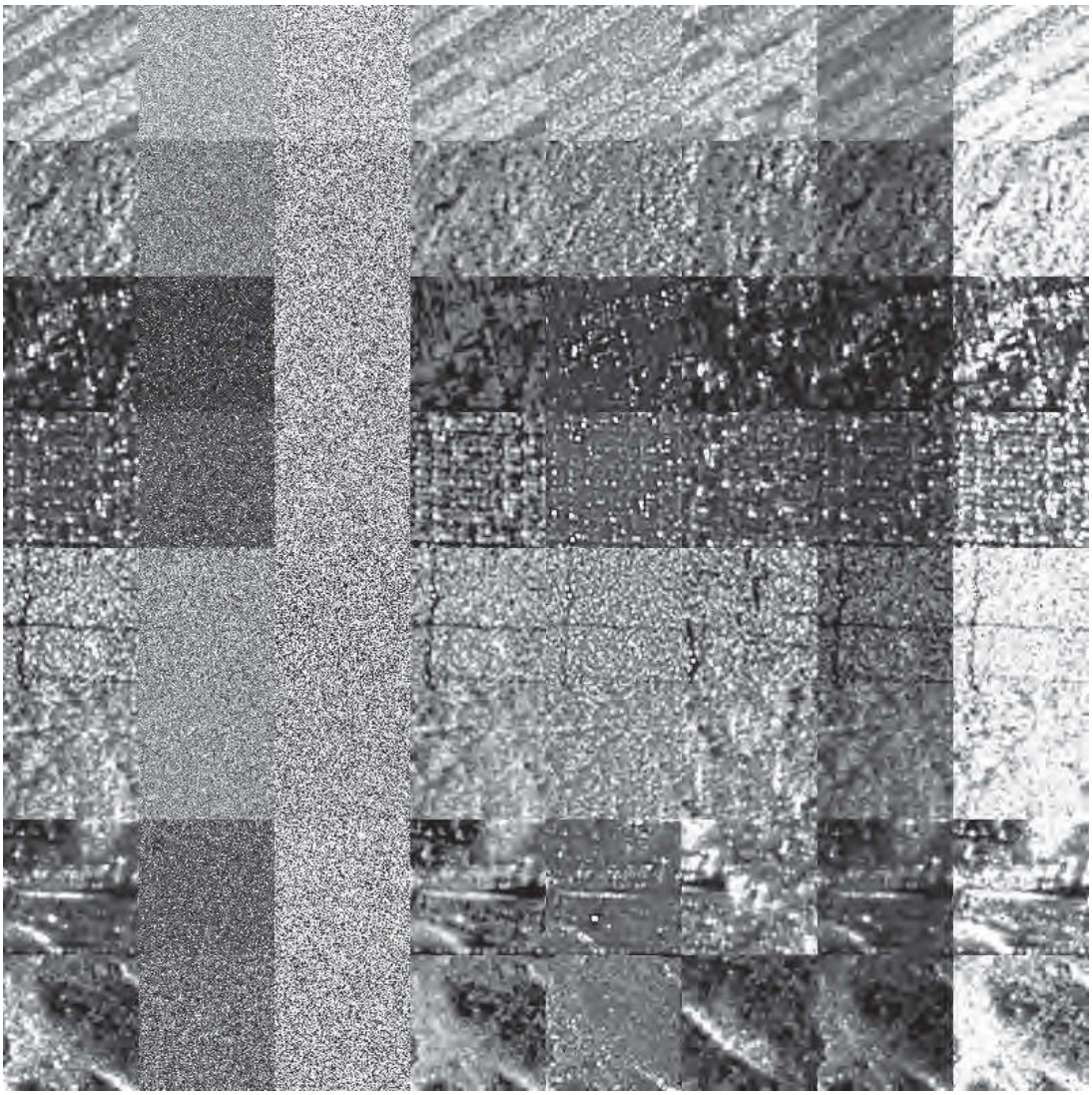


Figure 4.10: Change simulation of our eight classes. Each row contains a class (from top to bottom: agriculture, grass land, strongly reflecting buildings, dense buildings, fields, forest, industrial areas, and mountains). The first column is the patch for simulation selected from a large image. Starting from the second column, the simulated changes are random permutation, intensity change, first order statistical change, second order statistical change, texture change, and linear and nonlinear change.

4.3.6 Evaluation of Information Similarity Metrics for SAR Change Detection

4.3.6.1 Evaluation on Synthetic Data

To evaluate the performance of information similarity measures for change detection, synthetic data were generated by simulating statistical changes from eight image categories, i.e., agriculture, grass land, strongly reflecting buildings, dense buildings, field, forest, industrial area, and mountains. The image patches of the eight typical categories shown in the first column in Fig.

4.10 were selected and used for simulation. We assumed a $\mathcal{G}^0$ distribution in simulating first order statistical changes. The parameters are estimated using MoLC, where the equation is solved using the Levenberg-Marquardt algorithm. After transforming the data into a uniform distribution using the corresponding CDF, the mean value is kept for the Gaussian inverse CDF; however, the variance σ^2 is set to $0.6\sigma^2$. The new power slope α' in simulating the second order statistics is 0.6 for all simulations. The block size and overlap in the texture change simulation are respectively 10×10 and 4×4 pixels. Furthermore, the slope in our linear change simulation is set to 0.6. The quadratic function for the nonlinear simulation is $4(x - x_{min})^2/(x_{min} - x_{mean}) + x_{max}$, where x_{min}, x_{max}, and x_{mean} are respectively the minimum, the maximum and the mean value of the patch. To verify the data distribution for all seven kinds of changes, the data distributions of the last class (mountains), before and after the simulation, are shown in Fig. 4.11. The pixels on the diagonal are pixel pairs outside the simulated window. To test the robustness against noise, Gaussian noise with zero mean and a standard deviation of $\sigma = 4.0$ is added to the simulated images for all simulations.

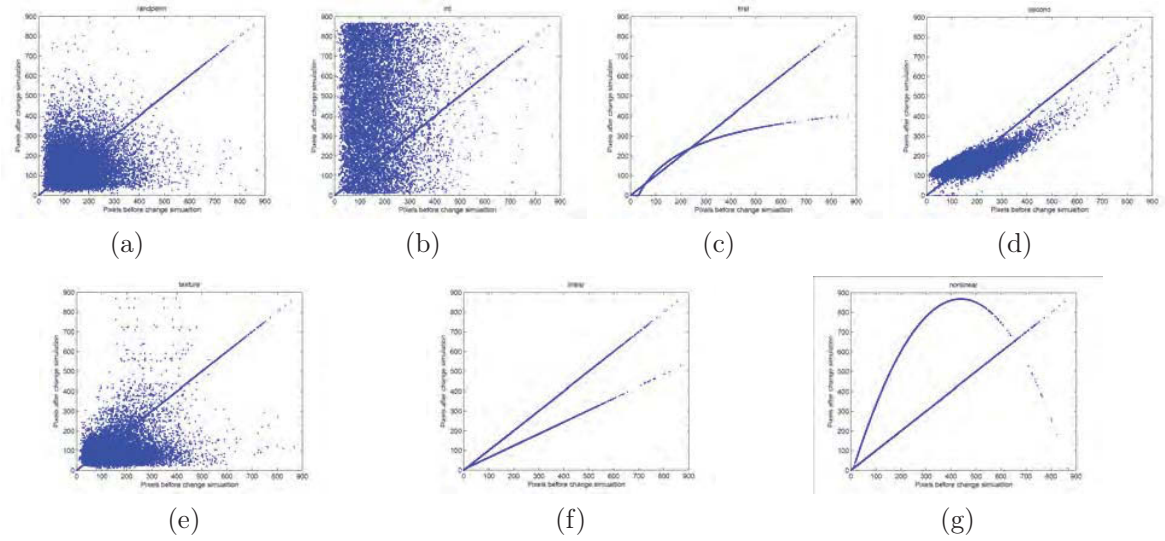
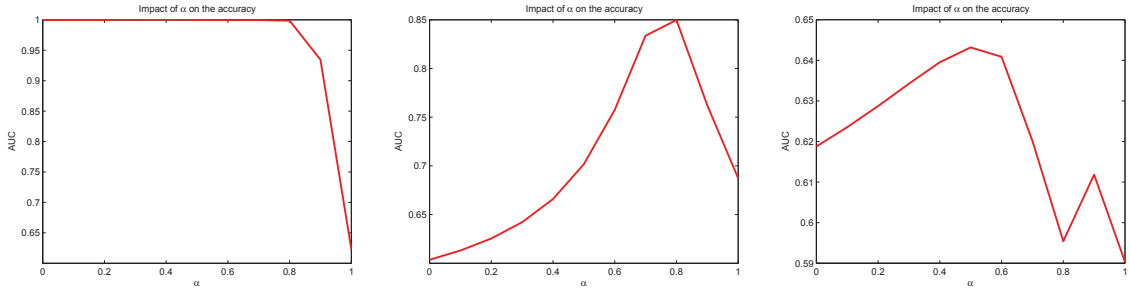


Figure 4.11: Data distribution of the seven simulations for mountains: (a) Random permutation; (b) Intensity change; (c) First order statistical change; (d) Second order statistical change; (e) Textural change; (f) Linear change; (g) Nonlinear change.

As the density estimation depends heavily on the sample size, we chose different window sizes varying from 9×9 to 23×23 for evaluation. Our results refer to the best performance using these window sizes. The joint distribution is estimated using KDE presented in section 4.1.6.2. All the simulated images are shown in Fig. 4.10. Visually, except for the first two kinds of changes, all the other simulated changes cannot be easily discriminated.

To evaluate the accuracy of the change map independently of the thresholding algorithm, the Receiver Operating Characteristic (ROC) curve is used and the Area Under ROC curve (AUC) is computed as a performance measure. The ROC curve can be considered as the evolution of the True Positive Rate (TPR) as a function of the False Alarm Rate (FAR). The TPR is defined as the fraction of correctly detected changes and the FPR is the fraction of correctly detected no changes. The area under the ROC curve is a good performance change indicator. The larger the area under the ROC curve, the better the performance of the method.

As a baseline for evaluation and comparison, the Kullback-Leibler divergence computed from two Gamma distributions and the method based on an Edgeworth series approximation described in Inglada & Mercier [2007] are also included in the experiments. The Gamma distribution is also estimated using MoLC, which is also solved by the Levenberg-Marquardt algorithm. All the information measures, i.e., the Kullback-Leibler divergence estimated respectively by an Edgeworth series, a Gamma distribution, a GΓD, and a $\mathcal{G}^0$ distribution as well as mutual information, variational information, and mixed information estimated by KDE, are applied to generate the change index. In the following sections, they are abbreviated respectively as KLD_EW, KLD_ΓD, KLD_GΓD, KLD_$\mathcal{G}^0$D, Mi_Info, Vi_Info, and Mix_Info. We categorize the information measures into two groups namely, Kullback-Leibler divergence and the other ones. We first analyze the performance of each group and then compare these two groups of information measures.



(a) (b) (c)

Figure 4.12: Impact of α on the performance of the mixed information measure for (a) intensity change, (b) first order change, and (c) nonlinear change in the case of agriculture.

Table 4.3: Accuracies of all information measures (AUC) on synthetic data.

Change	Class	KL_EW	KLD_ΓD	KLD_GΓD	KLD_$\mathcal{G}^0$D	Mi_Info	Vi_Info	Mix_Info
1	1	0.905869	0.995636	0.996085	0.896864	0.999887	0.847821	0.999922
	2	0.945882	0.995028	0.995046	0.855079	0.999934	0.894361	0.999937
	3	0.960234	0.991214	0.992547	0.995299	0.996962	0.797956	0.999510
	4	0.974829	0.994154	0.995433	0.992856	0.999801	0.881950	0.999868
	5	0.910498	0.996045	0.997156	0.845544	0.999950	0.956231	0.999952
	6	0.921725	0.995436	0.996561	0.853804	0.999924	0.831264	0.999928
	7	0.935217	0.996437	0.996663	0.993583	0.998694	0.452585	0.999673
	8	0.918851	0.997797	0.998195	0.984451	0.999847	0.684341	0.999899
Average		0.934138	0.995218	0.995961	0.927185	0.999375	0.793314	0.999836
2	1	0.999065	0.999408	0.999329	0.981883	0.999844	0.623255	0.999906
	2	0.997229	0.998601	0.997265	0.979202	0.999910	0.653480	0.999919
	3	0.991700	0.998385	0.995434	0.999540	0.995580	0.480794	0.999011
	4	0.995829	0.999639	0.998928	0.999682	0.999712	0.463737	0.999791
	5	0.999050	0.999456	0.999370	0.972855	0.999940	0.682438	0.999946
	6	0.997382	0.998701	0.998360	0.982934	0.999913	0.621072	0.999919
	7	0.995582	0.999196	0.997006	0.999058	0.997943	0.301998	0.999446
	8	0.996406	0.998837	0.997334	0.995067	0.999788	0.508549	0.999889
Average		0.996530	0.999028	0.997878	0.988778	0.999079	0.541915	0.999728
	1	0.939209	0.999483	0.999615	0.731370	0.699944	0.808941	0.961115
	2	0.959984	0.998347	0.998903	0.693797	0.648350	0.779886	0.960299

Continued on next page

3

Table 4.3 – continued from previous page

Change	Class	KL_EW	KLD_ΓD	KLD_GΓD	KLD_$\mathcal{G}^0$D	Mi_Info	Vi_Info	Mix_Info
	3	0.948976	0.997572	0.996972	0.999271	0.823282	0.784232	0.967266
	4	0.979731	0.998360	0.996400	0.997408	0.766588	0.744654	0.922876
	5	0.994141	0.999754	0.999815	0.782062	0.802711	0.853009	0.971172
	6	0.956423	0.999281	0.999433	0.816555	0.729312	0.749232	0.941863
	7	0.932264	0.997516	0.998229	0.975292	0.845928	0.418040	0.882035
	8	0.957234	0.999115	0.999285	0.881645	0.624618	0.635723	0.810482
Average		0.958495	0.998679	0.998581	0.859675	0.742592	0.721715	0.927138
4	1	0.921398	0.996228	0.997238	0.900639	0.999867	0.896399	0.999893
	2	0.919742	0.993117	0.994298	0.839860	0.999916	0.896315	0.999919
	3	0.935237	0.993159	0.995180	0.994504	0.997068	0.837598	0.999419
	4	0.892331	0.995995	0.996475	0.994928	0.999771	0.894391	0.999808
	5	0.905114	0.998305	0.998380	0.900438	0.999946	0.971848	0.999946
	6	0.905827	0.994940	0.996200	0.914227	0.999924	0.858817	0.999928
	7	0.954322	0.997219	0.997898	0.988916	0.998870	0.554673	0.999529
	8	0.965446	0.998272	0.998964	0.976700	0.999851	0.835585	0.999882
Average		0.924927	0.995905	0.996829	0.938776	0.999401	0.843203	0.999791
5	1	0.935773	0.995720	0.996671	0.849804	0.999879	0.911095	0.999913
	2	0.955804	0.996010	0.996269	0.832960	0.999942	0.933680	0.999943
	3	0.975641	0.988220	0.993042	0.995014	0.996803	0.845221	0.999255
	4	0.977483	0.993113	0.994581	0.991764	0.999816	0.891466	0.999844
	5	0.931464	0.996957	0.998009	0.864940	0.999951	0.972028	0.999952
	6	0.911134	0.994173	0.995305	0.897765	0.999926	0.862992	0.999927
	7	0.963926	0.992369	0.995030	0.993154	0.998385	0.531181	0.999698
	8	0.965250	0.992270	0.995201	0.972901	0.999922	0.877082	0.999928
Average		0.952059	0.993604	0.995513	0.924788	0.999327	0.853093	0.999807
6	1	0.997541	0.999802	0.999793	0.934891	0.729294	0.950479	0.988699
	2	0.999439	0.999812	0.999736	0.948039	0.672054	0.937091	0.984905
	3	0.999255	0.999821	0.999726	0.999595	0.685907	0.723192	0.975109
	4	0.999463	0.999823	0.999793	0.999624	0.720770	0.898044	0.978931
	5	0.999781	0.999917	0.999915	0.941271	0.834951	0.989265	0.992070
	6	0.999497	0.999817	0.999746	0.968870	0.753465	0.913314	0.983238
	7	0.996568	0.999722	0.999554	0.997934	0.803362	0.466225	0.975915
	8	0.998643	0.999828	0.999669	0.984473	0.637413	0.780493	0.987511
Average		0.998773	0.999818	0.999741	0.971837	0.729652	0.832263	0.983297
7	1	0.999310	0.999745	0.999384	0.959334	0.714786	0.591207	0.748268
	2	0.997543	0.999404	0.998320	0.990257	0.638822	0.506254	0.638822
	3	0.997864	0.999710	0.999095	0.999792	0.663316	0.383924	0.663316
	4	0.998259	0.999786	0.999635	0.999804	0.692234	0.289082	0.692234
	5	0.999333	0.999833	0.999700	0.960771	0.889819	0.849927	0.987758
	6	0.998038	0.999532	0.999180	0.991827	0.801424	0.778708	0.963212
	7	0.997814	0.999638	0.999016	0.998853	0.789627	0.232296	0.789627
	8	0.998285	0.999500	0.998947	0.983555	0.604202	0.355492	0.604202
Average		0.998305	0.999643	0.999160	0.985524	0.724279	0.498361	0.760930

The accuracy in terms of AUC is presented in Table 4.3. From top to bottom, the change index $(1, 2, ..., 7)$ in the first column is random permutation followed by intensity change, first order statistical change, second order statistical change, texture change, linear change, and nonlinear change respectively. The class index $(1, 2, ..., 8)$ in the second column is agriculture, grass land, strongly reflecting buildings, dense buildings, field, forest, industrial area, and mountains. The best accuracy is marked in red. It is clear that Mix_Info always performs well for random intensity changes, second order changes and texture changes. For these three kinds of changes, Mi_Info has a similar performance compared with Mix_Info. In contrast, Vi_Info performs quite badly. The only difference between these three information measures is their parameter α. In order

to find the reason for it, the impact of α on the accuracy for the random intensity changes in the first class (agriculture) is plotted in Fig. 4.12(a). In this case, the performance of Mix_Info remains stable at 0.999832 when α is smaller than 0.8. Then, the accuracy decreases significantly and reach the very low value of 0.623260. This pattern remains the same for second order and texture changes.

For first order and linear changes, there is a big improvement of Mix_Info compared with Mi_Info and Vi_Info because the maximum accuracy is reached in the middle of the interval of $\alpha \in (0.0, 1.0)$. The impact of α on the accuracy for first order changes in agriculture is shown in Fig. 4.12(b). In this case, the accuracy increases as α increases and reaches its peak when $\alpha = 0.8$. Beyond the peak, the accuracy decreases dramatically. That is why the accuracies of both mutual information and variational information are quite low for first order and linear changes. In this case, we conclude that Mix_Info is a better choice compared to Mi_Info and Vi_Info for the detection of first order and linear changes if the α parameter can be selected appropriately.

Unfortunately, all these three information measures perform quite badly for nonlinear changes. The impact of α on nonlinear changes is shown in Fig. 4.12(c). In this case, the accuracy fluctuates between $\alpha = 0.8$ and $\alpha = 0.9$. However, the attainable accuracy is still very low.

As for the Kullback-Leibler divergence, both KLD_ΓD and KLD_GΓD perform quite well with an accuracy of better than 99% for almost all kinds of changes in all classes. For random permutations and all three kinds of statistical changes, KLD_GΓD performs slightly better than KLD_ΓD while it performs slightly worse for intensity changes and linear and nonlinear changes. This result confirms that both the Gamma distribution and GΓD are accurate models for fully developed speckle SAR image modeling. Although the Edgeworth series expansion is not specially designed for SAR image modeling, KLD_EW performs still well in spite of its slightly worse accuracy compared with KLD_ΓD and KLD_GΓD. Although the overall accuracy of KLD_$\mathcal{G}^0$D is not high, it is worth to note that it performs quite well for urban categories, like the third (strongly reflecting buildings), fourth (dense buildings) and seventh categories (industrial areas). This result confirms the conclusions made by Tison *et al.* [2004] Frery *et al.* [1997] that the $\mathcal{G}^0$ model (or Fisher model) is an efficient model for urban areas in high-resolution SAR images.

In spite of the superiority of Mix_Info compared with Mi_Info and Vi_Info, its performance is still worse than KLD_ΓD for first order and linear changes. However, the performances of both Mix_Info and Mi_Info are better than the Kullback-Leibler divergence for random intensity changes, second order and texture changes. For first order, linear and nonlinear changes, the Kullback-Leibler divergence performs better than the other three information measures. This is because the Kullback-Leibler divergence describes the distance between the marginal distributions, which contains only the first order information. On the contrary, the other three information measures consider not only the first order distribution, but also the second order (joint) distribution.

Among the parametric models used in computing the Kullback-Leibler divergence, GΓD performs generally better than other models. In addition, the Gamma distribution is also very good for texture classes, like grass land and agriculture. We can also observe that the $\mathcal{G}^0$ distribution performs well for urban areas, like densely built-up areas, areas with strongly reflecting buildings and industrial areas. Nevertheless, the $\mathcal{G}^0$ distribution is worse in the case of areas rich in texture when compared with the Gamma or GΓD distribution.

From the evaluation of information measures on the synthetic data, we conclude that Kullback-Leibler performs quite well for intensity changes associated with first order statistics. On the other hand, both mutual information and mixed information are better alternatives for detecting changes in second and higher order statistics.

4.3.6.2 Evaluation on Real SAR Datasets

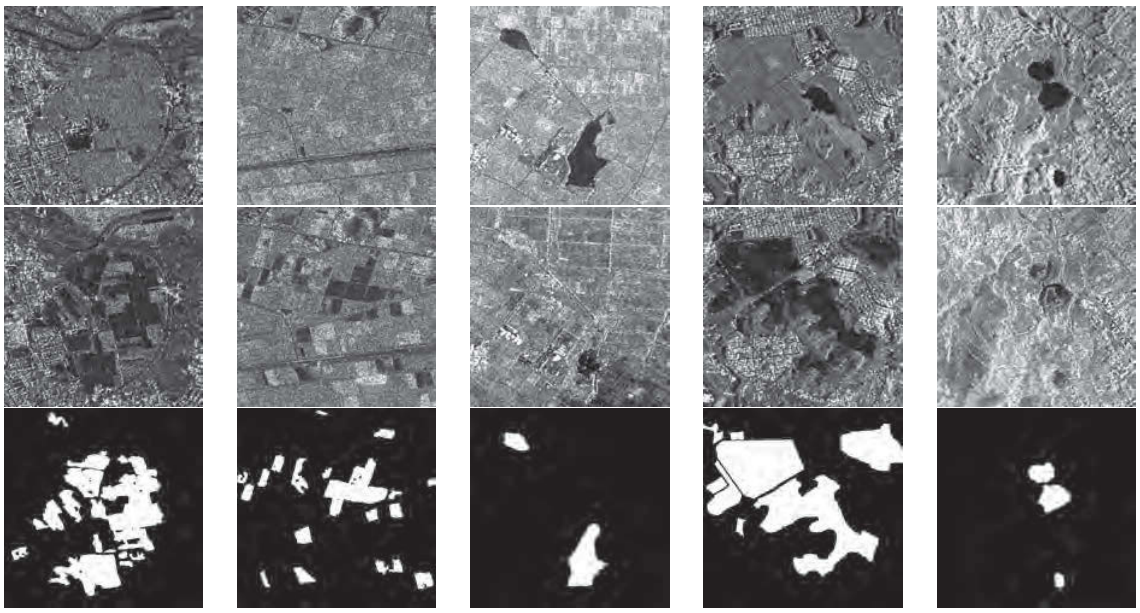


Figure 4.13: TerraSAR-X datasets being used for evaluation. The first row contains the original images, while the second row contains the images with changes. The third row lists the reference data generated by manual interpretation.

We selected five datasets of TerraSAR-X images corresponding to intensity changes and statistical changes. The first four datasets are selected as shown in the first two rows in Fig. 4.13 from two radiometrically enhanced TerraSAR-X images acquired in strip map mode prior to (on Oct. 20. 2010) and after (on May 6, 2011) the Sendai earthquake in Japan. Their pixels spacing is about 2.5 m. The sizes of these four image pairs are respectively 549×560, 613×641, 590×687, and 689×734 pixels. As the two TerraSAR-X images acquired before and after the disaster have nearly the same imaging parameters, we can reasonably assume that there is only a linear geometrical translation between the images. To achieve precise registration, ten strong point scatterers from each image were manually selected to determine the translation along both azimuth and range direction. To make sure that the translation is precise, the normalized cross-correlation is computed to check the translation and the residual pixel shift is less than one pixel. The reference data shown in the third row were produced through careful manual interpretation by referring to optical images. Due to the earthquake, a Tsunami happened, which led to devastating flooding, as can be seen from the images. In the first two datasets, the agricultural fields were severely flooded and the intensities were dramatically changed; these are appropriate scenarios for performance assessment in detecting intensity changes. The third and fourth datasets contain both intensity and statistical changes, which were used for assessing statistical changes. The fifth dataset is selected from another TerraSAR-X image covering the Vâlcea county in Romania. The images have a pixel spacing of 2.5 m and an incidence angle of around 36^{o}. The image registration was done in the same manner. In this scenario, some under water grass and vegetation in the three lakes were growing, which changes the statistical characteristics of the images.

The experimental setup for evaluation follows the evaluation of the synthetic dataset. Apart

from the conclusions drawn in the previous evaluation, here we also assess the impact of the window size on the accuracy of change detection. In the following sections, a change index map is shown using color for highlighting.

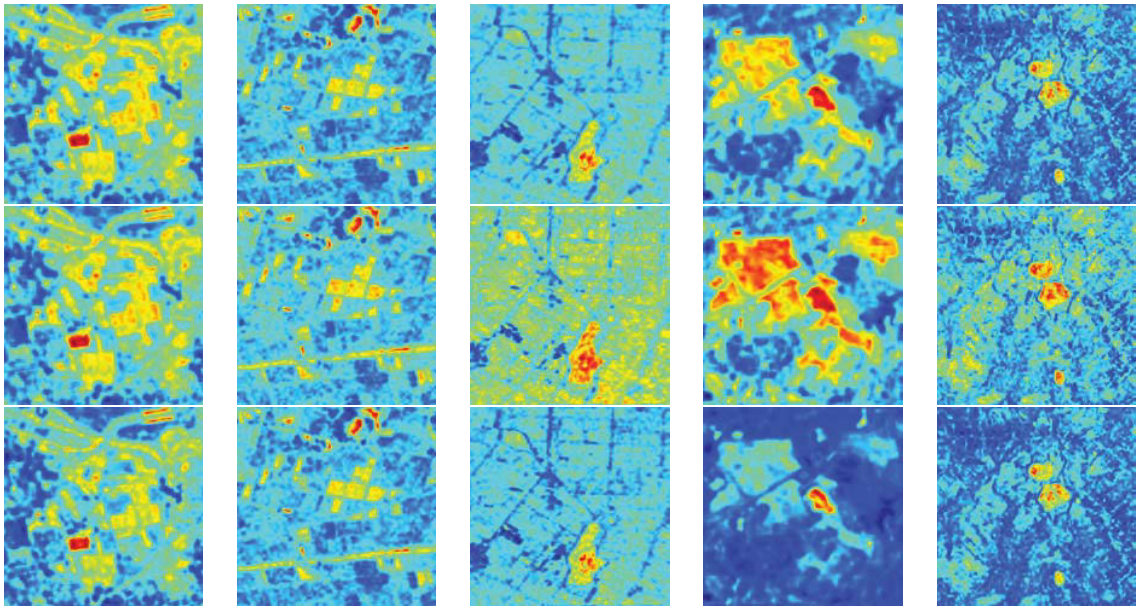


Figure 4.14: The best change index map (in terms of AUC) of the five datasets delivered by Mix_Info, Mi_Info and Vi_Info (row by row form top to bottom). The corresponding window sizes can be found from Table 4.4. Note that the change index maps in the second row are inverted such that change pixels have large values.

The AUCs of all the information measures on the five datasets are presented in Table 4.4. In contrast to the evaluation on synthetic data, for the same window size, all the three information measures Mix_Info, Mi_Info and Vi_Info have similar performance, although Mix_Info is slightly better. The impact of α for the first three datasets are shown in Fig. 4.16 and the best change index maps of the five datasets created by Mix_Info, Mi_Info and Vi_Info are shown in Fig. 4.14. This result leads to the conclusion that all the three information measures can be applied to change detection. Altogether, by comparing the performance on the first two datasets and the performance on the remaining three datasets, we can see they are much better at detecting statistical changes than intensity changes.

In contrast to these three information measurers, the Kullback-Leibler divergence has a better performance for the first two datasets, which mainly consist of intensity changes. For the last three datasets, especially the third and the fifth ones, the Kullback-Leibler divergence performs worse with an accuracy of around 80% for the third dataset and 70% for the fifth dataset. This result confirms the conclusion drawn from the evaluation on synthetic data that Kullback-Leibler performs well for intensity changes and both mutual information and mixed information are better alternatives for changes in second and higher order statistics.

As for the parametric models used in computing the Kullback-Leibler divergence, all the three models, i.e., Gamma, GΓD, and $\mathcal{G}^0$, have similar performance. For example, in the case of the second and the fourth dataset, the best change index maps (in terms of AUC) of KLD_EW, KLD_ΓD, KLD_GΓD, and KLD_$\mathcal{G}^0$D are shown in Fig. 4.15. However, as can be seen from the first column in Fig. 4.15, there are many false alarm detections and less true detections.

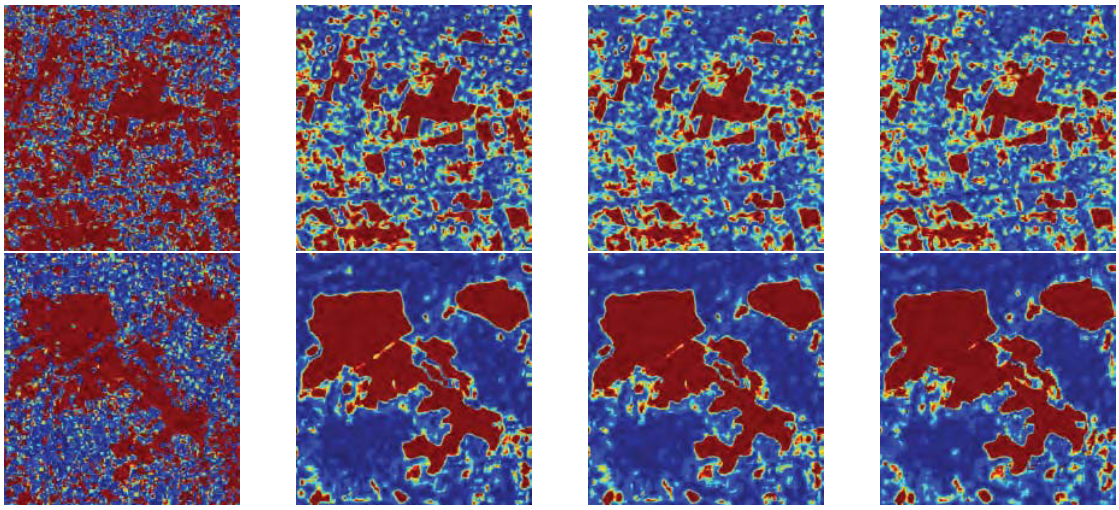
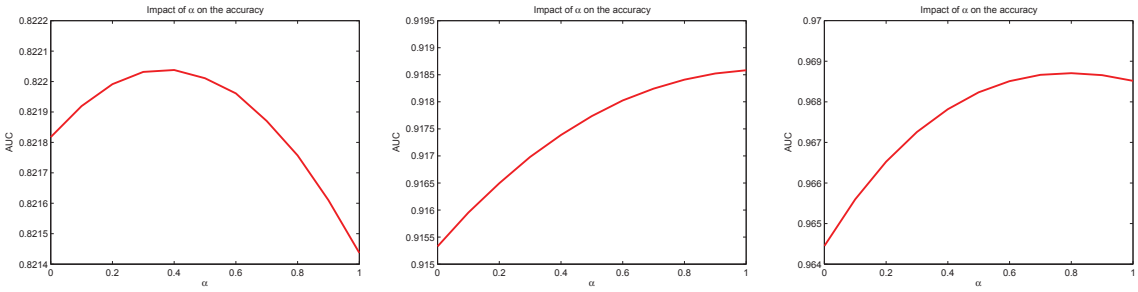


Figure 4.15: Change index maps of the second (first row) and the fourth (second row) datasets delivered by different Kullback-Leibler divergence calculations (from left to right) KLD_EW, KLD_ΓD, KLD_GΓD, and KLD_$\mathcal{G}^0$D. The corresponding window sizes can be found from Table. 4.4.



(a) (b) (c)

Figure 4.16: Impact of α on the performance of mixed information measures for real datasets: (a) our first dataset (α has a similar impact on the fourth data set); (b) our second dataset; (c) our third dataset (α has a similar impact on the fifth data set).

The reason is that KLD_EW relies on an Edgeworth series expansion, which is based on the assumption that the distribution is close to a Gaussian, which might not be the case for SAR images. Furthermore, a Hermite polynomial is used for series expansion that has fluctuating characteristics Zheng & You [2013], which can introduce false detections. In particular, please note that there are some missing points in the change index maps of KLD_$\mathcal{G}^0$D in the fourth column. For these pixels, no solution can be found. Local interpolation is applied to find the missing values. However, the change index maps delivered by both KLD_ΓD and KLD_GΓD do not have missing values, which confirms their better convergence ability in solving the MolC equations by the Levenberg-Marquardt algorithm.

In addition, it seems that KLD_EW has a better performance with smaller windows that degrades when the window size increases. For the second and fifth datasets, KLD_EW achieves its best performance with a window size of 9×9 pixels. Similarly, it achieves its maximum

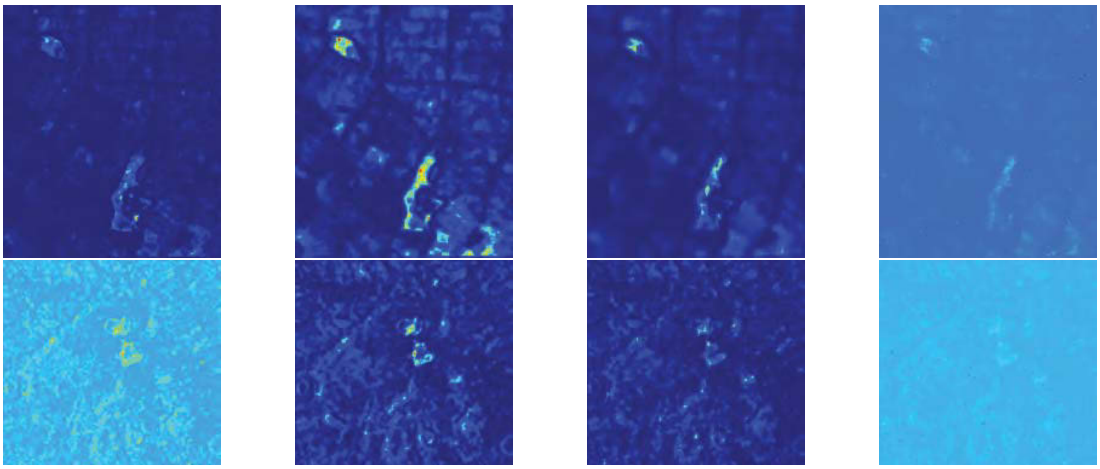


Figure 4.17: Change index maps of the third (first row) and the fifth (second row) datasets delivered by different Kullback-Leibler divergence calculations (from left to right) KLD_EW, KLD_ΓD, KLD_GΓD, and KLD_$\mathcal{G}^0$D. The corresponding window sizes can be found from Table. 4.4.

accuracy for the first and third datasets with a window size of 11×11 pixels, Beyond this value, the accuracy decreases when the window size decreases. The second group of information measures achieves its best accuracies using a medium window size of around 15×15 pixels. Through the evaluation on the real dataset, the conclusions drawn from the experiments using synthetic data have been confirmed.

Table 4.4: Accuracies of all information measures (AUC) on the real data sets.

Data	Window	KL_EW	KLD_ΓD	KLD_GΓD	KLD_$\mathcal{G}^0$D	Mi_Info	Vi_Info	Mix_Info
1	9	0.922068	0.971424	0.971753	0.970698	0.821815	0.821442	0.822040
	11	0.924071	0.976756	0.977358	0.976784	0.837129	0.835965	0.837129
	13	0.922744	0.979145	0.979886	0.979857	0.844238	0.842517	0.844238
	15	0.919599	0.980011	0.980939	0.981249	0.847117	0.844828	0.847117
	17	0.915760	0.979895	0.981023	0.981758	0.847582	0.844647	0.847582
	19	0.911500	0.979177	0.980567	0.981216	0.847003	0.843446	0.847003
	21	0.907207	0.978107	0.979816	0.980315	0.845791	0.841778	0.845791
	23	0.902817	0.976889	0.978866	0.979387	0.844214	0.839835	0.844215
Average		0.915721	0.977675	0.978776	0.978908	0.841861	0.839307	0.841889
2	9	0.929449	0.969822	0.966110	0.965310	0.915330	0.918586	0.918586
	11	0.924719	0.972097	0.969164	0.968414	0.926902	0.929379	0.929379
	13	0.917776	0.972337	0.970001	0.969148	0.930301	0.932548	0.932548
	15	0.909698	0.971331	0.969536	0.968981	0.929982	0.932291	0.932291
	17	0.901234	0.969642	0.968409	0.968018	0.928055	0.930409	0.930409
	19	0.892929	0.967502	0.966674	0.966218	0.925478	0.928040	0.928040
	21	0.884759	0.965082	0.964571	0.964720	0.922645	0.925379	0.925379
	23	0.876666	0.962649	0.962354	0.962953	0.919741	0.922684	0.922684
Average		0.904654	0.968808	0.967102	0.966720	0.924804	0.927415	0.927415
3	9	0.823141	0.800394	0.798198	0.779554	0.964448	0.968516	0.968713
	11	0.827124	0.803911	0.807116	0.788538	0.964739	0.969374	0.969398
	13	0.826937	0.805454	0.811722	0.790312	0.957825	0.964668	0.964668
	15	0.823118	0.805755	0.814481	0.781796	0.947473	0.957384	0.957384
	17	0.816634	0.805201	0.815944	0.782544	0.935168	0.949237	0.949237

Continued on next page

Table 4.4 – continued from previous page

Data set	Window	KL_EW	KLD_ΓD	KLD_GΓD	KLD_$\mathcal{G}^0$D	Mi_Info	Vi_Info	Mix_Info
	19	0.807750	0.803838	0.817254	0.792693	0.921571	0.940724	0.940724
	21	0.796203	0.801247	0.817838	0.789024	0.907209	0.932472	0.932472
	23	0.782582	0.797510	0.815866	0.778318	0.892231	0.924318	0.924318
Average		0.812936	0.802914	0.812302	0.785347	0.936333	0.950837	0.950864
4	9	0.843292	0.920941	0.810977	0.901906	0.922262	0.921692	0.922481
	11	0.844661	0.928675	0.909291	0.913424	0.934846	0.933540	0.934859
	13	0.847424	0.934508	0.914157	0.920230	0.943000	0.941328	0.943000
	15	0.847215	0.939361	0.930109	0.925439	0.948644	0.946840	0.948644
	17	0.845991	0.943153	0.941859	0.928885	0.952724	0.950896	0.952724
	19	0.844842	0.946100	0.944333	0.929443	0.955686	0.953944	0.955699
	21	0.841318	0.948468	0.949601	0.931992	0.957875	0.956329	0.957970
	23	0.836021	0.950413	0.951255	0.935897	0.959478	0.958205	0.959755
Average		0.843845	0.938952	0.918948	0.923402	0.946814	0.945347	0.946891
5	9	0.722352	0.718941	0.687349	0.703202	0.993849	0.995253	0.995331
	11	0.717181	0.724957	0.700668	0.713208	0.992772	0.994946	0.994946
	13	0.704665	0.725347	0.707985	0.719949	0.989162	0.993211	0.993211
	15	0.687349	0.721120	0.709587	0.721673	0.984428	0.991116	0.991116
	17	0.665274	0.712315	0.706271	0.717078	0.978931	0.988943	0.988943
	19	0.639567	0.699467	0.699216	0.711239	0.972611	0.986825	0.986825
	21	0.612778	0.684094	0.689286	0.703597	0.965344	0.984850	0.984850
	23	0.586215	0.667314	0.679796	0.694207	0.957369	0.983242	0.983242
Average		0.666923	0.706694	0.697520	0.710519	0.979308	0.989798	0.989808

4.3.7 Summary and Discussion

A benchmark dataset is created for the purpose of evaluation. A comprehensive evaluation of information similarity measures, i.e., Kullback-Leibler divergence, mutual information, variational information, and mixed information for SAR change detection is carried out based on a synthetic dataset and a real dataset. Closed form expressions of the Kullback-Leibler divergence computed from two parametric models, i.e., GΓD and Fisher ($\mathcal{G}^0$) distribution, are given, which speeds up the computation of change detections. In addition, the impact of the α parameter in mixed information on the change detection accuracy is also analyzed and discussed. The Gamma distribution, GΓD, and the $\mathcal{G}^0$ (or Fisher) distribution are investigated for estimating the Kullback-Leibler divergence. All these three models perform well for SAR change detection using the Kullback-Leibler divergence. The $\mathcal{G}^0$ (or Fisher) distribution performs particularly better in urban areas.

Through the evaluation of information measures on both synthetic and real data, we conclude that Kullback-Leibler performs quite well in detecting changes in intensity. In contrast, both mutual, variational, and mixed information are better alternatives for changes in second order and high order statistics. In addition, the impact of the α parameter for different types of changes and categories has been analyzed experimentally.

4.4 SAR Image Change Detection in the Wavelet Domain

In this section, we turn to the wavelet domain and investigate the possibility of modeling wavelet coefficients for SAR change detection because texture can be captured efficiently by a statistical modeling of the wavelet coefficients.

4.4.1 Wavelet Coefficient Modeling

We introduce two statistical models (GGD and GΓD) for the modeling of wavelet sub-band coefficients. As we use sliding windows for change detection, the parameter estimation methods should be fast enough. Fast parameter estimations of these models are investigated and detailed in the wavelet domain.

4.4.1.1 Generalized Gaussian Distribution (GGD)

It has been widely acknowledged that the distribution of wavelet coefficients can be accurately modeled by GGD Mallat [1989], Sharifi & Leon-Garcia [1995], which is given as follows

$$p_X(x,\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)} \exp\Big(-|\frac{x}{\alpha}|\Big)^{\beta}, \quad \alpha,\beta>0 \tag{4.88}$$

where α is a scale parameter, β is a shape parameter, and Γ denotes the gamma function. The corresponding CDF is given as

$$F_X(x,\alpha,\beta) = \mathrm{sgn}(x)\frac{\gamma\Big(1/\beta, |\frac{x}{\alpha}|^{\beta}\Big)}{2\Gamma(1/\beta)} + \frac{1}{2} \tag{4.89}$$

where γ is the lower incomplete gamma function defined as $\gamma(a,x) = \int_0^x t^{a-1}e^{-t}dt$ and sgn(x) is the sign function. Due to the difficulties in applying MLE to parameter estimation, an alternative and fast method for estimating the shape parameter based on a convex shape equation was proposed recently by Song [2006]. The uniqueness and the consistency of the method have been proven theoretically, even in the case when the initial value is far away from the true value. By simple mathematical manipulation, one can show that a random variable $Y = |X/\alpha|^{\beta}$ follows gamma distribution

$$F_Y(y) = \frac{y^{\frac{1}{\beta}-1}e^{-y}}{\Gamma(1/\beta)} \tag{4.90}$$

Therefore, the first two moments of Y can be given by $E(Y) = 1/\beta$ and $E(Y^2) = (1+\beta)/\beta^2$. Following the same idea of Mallat's ratio Mallat [1989], a new ratio can be defined as

$$\frac{E(Y^2)}{E^2(Y)} = \frac{E\Big(|\frac{X}{\alpha}|^{2\beta}\Big)}{E^2\Big(|\frac{X}{\alpha}|\Big)} = \beta + 1 \tag{4.91}$$

Thus, the shape parameter β can be considered as the root of Eq. (4.92) as the ratio can be estimated by samples.

$$Q(\beta) = \frac{E\Big(|X|^{2\beta}\Big)}{\Big(E(|X|^{\beta})\Big)^2} - \beta - 1 \tag{4.92}$$

It has been demonstrated that this equation has a unique global root on the positive real axis and any initial value in a semi-infinite interval will converge to the true value. Based on this theoretical guarantee, the root can be solved quickly by a Newton-Raphson approach using the iterative equation (4.93).

$$\beta_{i+1} = \beta_i - \frac{Q(\beta_i)}{Q'(\beta_i)} \tag{4.93}$$

where $Q'(\beta)$ is defined as the derivative of $Q(\beta)$, given as follows

$$Q'(\beta) = \frac{2E\Big(|X|^{2\beta}\log(|X|)\Big)}{E^2\Big(|X|^{\beta}\Big)} - \frac{2E\Big(|X|^{\beta}\log(|X|)\Big)E\Big(|X|^{2\beta}\Big)}{E^3\Big(|X|^{\beta}\Big)} - 1 \tag{4.94}$$

In Eq. (4.94), the expectations need to be replaced by sample estimations in all iterations. One advantage of this method is that it does not depend on a gamma or polygamma function, which decreases the computational complexity and leads to a fast estimation of the shape parameter.

4.4.1.2 Generalized Gamma Distribution (GΓD)

Recently, it has been observed that GGD cannot fully characterized the wavelet coefficient distribution of a complex scene with high variations Choy & Tong [2010], Van de Wouwer *et al.* [1999], Song [2008]. As an alternative, GΓD is a promising option with nice mathematical properties over GGD that has been proposed for coefficient modeling. GΓD has been introduced in section 4.1.3. Besides MoLC for parameter estimation, one can use Scale-Independent Shape Estimation (SISE) for fast parameter estimation Song [2008], where the uniqueness and consistency of the solution have been proven mathematically. Following the same idea for the estimation of the GGD parameters, a new random variable is defined as $Y = (X/\alpha)^{\beta}$ having a gamma distribution

$$p_Y(y) = \frac{y^{\lambda}\exp(-y)}{\Gamma(\lambda)} \tag{4.95}$$

Similarly, the ratio of the second moment to the square of the first moment of the random variable Y is given as

$$\frac{E(X^{2\beta})}{\Big(E(X^{\beta})\Big)^2} = 1 + \frac{1}{\lambda} \tag{4.96}$$

In contrast to Eq. (4.91), both the shape parameter β and the index shape parameter λ are involved in (4.96) and have to be estimated. To remove the index shape parameter λ from Eq. (4.96), it can be expressed by a scale independent function of the shape parameter β. Let the mapping M be defined as $M(t) : t \mapsto E(X^t)$ and the derivative with respect to t defined by $M'(t) : t \mapsto E(X^t\log(X))$. In addition, an auxiliary mapping is defined as $R'(t) = M'(t)/M(t)$, thus $R'(0) = \lim_{t\to x^+} R'(t)$. Based on these definitions, Eq. (4.96) can be rewritten as

$$\log M(2\beta) - 2\log M(\beta) = \log\Big(1 + \frac{1}{\lambda}\Big) \tag{4.97}$$

With the density function in Eq. (4.24), $M(t)$ and $M'(t)$ can be induced respectively as

$$M(t) = \frac{\alpha^t\Gamma(\lambda + t\beta^{-1})}{\Gamma(\lambda)} \quad M'(t) = M(t)\Big(\log\alpha + \frac{\Psi(\lambda + t\beta^{-1})}{\beta}\Big) \tag{4.98}$$

With (4.98) and $\Psi(0, 1+x) = \Psi(0, x) + 1/x$, $R'(t)$ can be evaluated as

$$R'(\beta) - R'(0) = \frac{\Psi(0, \lambda+1) - \Psi(0, \lambda)}{\beta} = \frac{1}{\beta\lambda} \tag{4.99}$$

Therefore,

$$\frac{1}{\lambda} = \beta\Big(R'(\beta) - R'(0)\Big) \tag{4.100}$$

Substituting Eq. (4.100) into Eq. (4.97), the scale independent shape equation can be derived as

$$\log M(2\beta) - 2\log M(\beta) - \log\Big(1 + \beta R'(\beta) - \beta R'(0)\Big) = 0 \tag{4.101}$$

Eq. (4.101) does not involve the parameter λ, which can be solved for the shape parameter β. Within the shape equation (4.101), the expectation values and the derivatives are unknown and can be estimated by samples. To this end, we define the sample estimation as follows

$$\begin{aligned} M_N(t) &= \frac{1}{N}\sum_{i=1}^{N} X_i^t \quad M'_N(t) = \frac{1}{N}\sum_{i=1}^{N} X_i^t \log X_i \\ R'_N(t) &= \frac{M'_N(t)}{M_N(t)} \quad R'_N(0) = \frac{1}{N}\sum_{i=1}^{N} \log(X_i) \end{aligned} \tag{4.102}$$

Replacing the theoretical expectation value and derivatives with a sample estimation, we obtain the sample SISE

$$S_N(\beta) = \log M_N(2\beta) - 2\log M_N(\beta) - \log\Big(1 + \beta R'_N(\beta) - \beta R'_N(0)\Big) \tag{4.103}$$

Using numerical iterative methods, such as the Newton-Raphson algorithm, the shape parameter can be solved iteratively by

$$\hat{\beta}_{i+1} = \hat{\beta}_i - \frac{S_N(\hat{\beta}_i)}{S'_N(\hat{\beta}_i)} \tag{4.104}$$

where $S'_N(\hat{\beta}_i)$ is the derivative of $S_N(\hat{\beta}_i)$.

Similarly, the Kullback-Leibler divergence between two GGDs shown in Eq. (4.105) and GΓDs shown in Eq. (4.87) are used for similarity assessment.

$$KLD_{GGD}\Big(p(x:\alpha_1,\beta_1)||p(x:\alpha_2,\beta_2)\Big) = \log\Big(\frac{\beta_1\alpha_2\Gamma(1/\beta_2)}{\beta_2\alpha_1\Gamma(1/\beta_1)}\Big) + \Big(\frac{\alpha_1}{\alpha_2}\Big)^{\beta_2}\frac{\Gamma\Big((\beta_2+1)/\beta_2\Big)}{\Gamma(1/\beta_1)} - \frac{1}{\beta_1} \tag{4.105}$$

To exploit the multiscale property of the wavelet transform, we assume that the subbands are independent such that the total similarity of two sliding windows is defined as the sum of the similarity measures of each subband in Eq. (4.106) and Eq. (4.107).

$$TKLD_{GGD}(I_1||I_2) = \sum_{i=1}^{3L} KLD_{GGD}\Big(p^i(x:\alpha_1,\beta_1)||p^i(x:\alpha_2,\beta_2)\Big) \tag{4.106}$$

$$TKLD_{\mathrm{G\Gamma D}}(I_1||I_2) = \sum_{i=1}^{L} KLD_{\mathrm{G\Gamma D}}\Big(p^i(x;\alpha_1,\beta_1,\lambda_1)||p^i(x;\alpha_2,\beta_2,\lambda_2)\Big) \tag{4.107}$$

I_1 and I_2 are the two sliding windows being used for comparison. L is the number of scales. p^i is the estimated distribution of the wavelet coefficients at scale i.

4.4.2 Experiments and Evaluation

4.4.2.1 Datasets and Experimental Settings

To evaluate the proposed method, three datasets shown in Fig. 4.18 were selected from two radiometrically enhanced TerraSAR-X images acquired in strip map mode prior to (on Oct. 20,

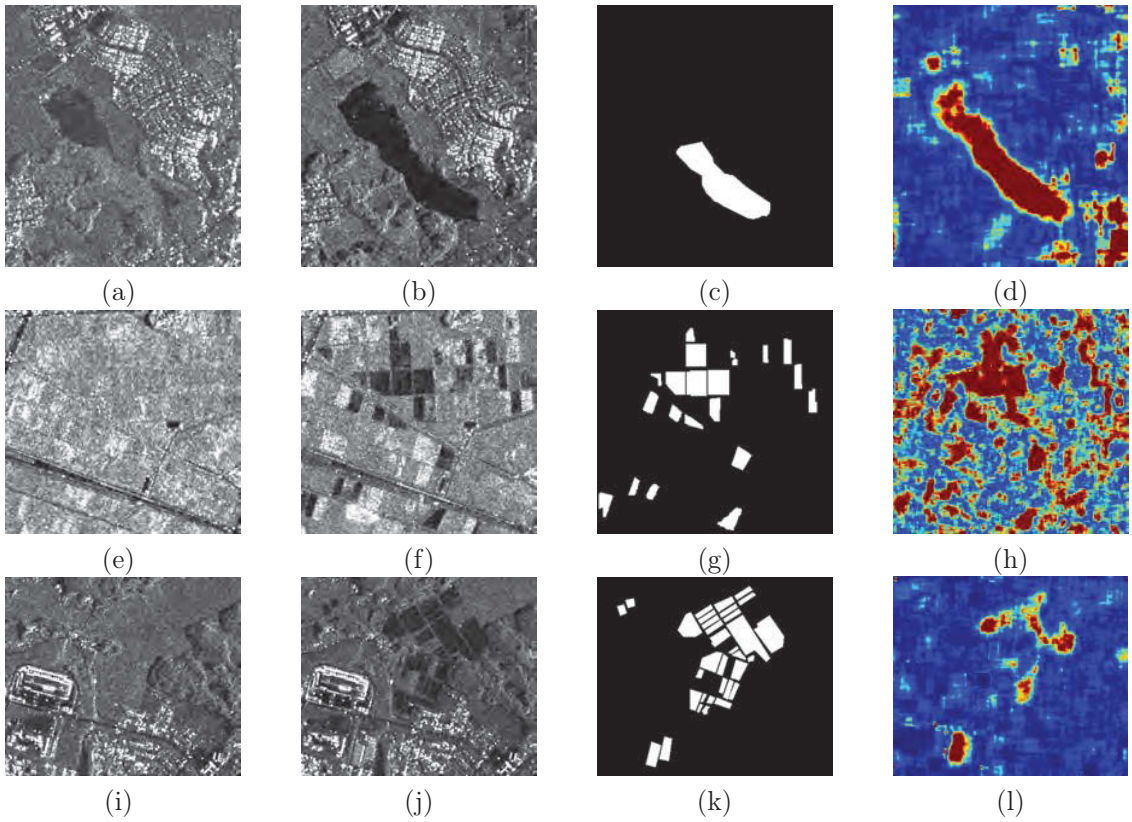


Figure 4.18: The three datasets used for evaluation are selected from two TerraSAR-X images acquired in Sentai before and after Tsunami. The first column shows the image before the disaster and the second column shows the images after the disaster. The third column is the reference map produced by manual interpretation. The last column shows the best change maps.

2010) and after (on May 6, 2011) the Sendai earthquake in Japan. Their pixel spacing is about 1.9 m. In the first test scenario, the area of the visible lake was changed due to heavy flooding when comparing Fig. 4.18(a) and (b). In the second and the third test areas (Fig. 4.18(e)(f) and (i)(j)), some agricultural fields were flooded and full of water as shown in Fig. 4.18. The reference change map of the three test sites on the right side were produced manually by referencing optical images taken after the disaster.

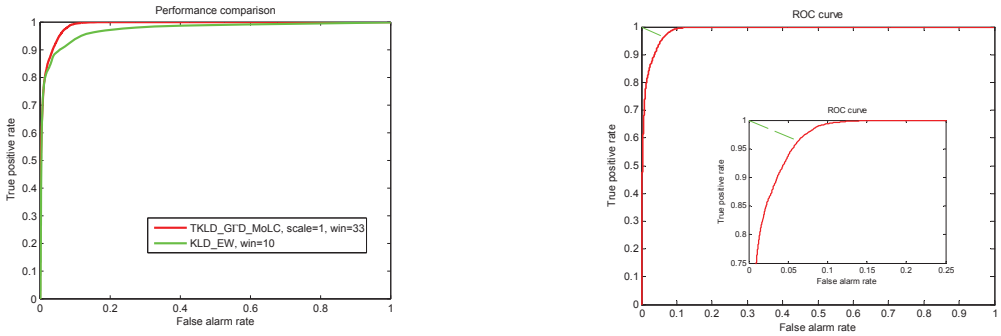
Different scales and window sizes are used for a performance evaluation using the three datasets. Each sliding window with a size ranging from 10×10 to 33×33 is decomposed into $L = (1, 2, 3)$ scales using an Undecimated Wavelet Transform (UWT) with a Daubechies filter bank Strang & Nguyen [1997]. The GGD and GΓD models are estimated at each scale and with different window sizes. At each scale, the GGD model was estimated by maximum likelihood and the shape equation, while the GΓD model was estimated by MoLC and the shape equation, and then were applied to change detection (abbreviated as TKLD_GGD_MLE, TKLD_GGD_SE, TKLD_GΓD_SE, and TKLD_GΓD_MoLC in the following tables). The change indicators of each scale are summed up by Eq. (4.106) and Eq. (4.107) as the final change map.

AUC is used as a performance measure. Our focus is to generate accurate change maps, rather than a binary change map, as the definition of change depends heavily on applications. However, to allow a comparison with other methods, an optimal threshold corresponding to the

point nearest to (0.0, 1.0) on the ROC curve as shown in Fig. 4.19(b) is selected which optimizes the performance. Based on the optimal threshold, the TPR and FPR are also reported.

For the purpose of comparison, we selected a method performed in the spatial domain Inglada & Mercier [2007] and another method in the wavelet domain Çelik [2009]. The second one clusters the wavelet coefficients into two classes associated with change and no-change. It does not generate a quantitative change map but only a final binary change map. In this case, we cannot compute the ROC curve. The method can only be compared in terms of TPR and FAR. In all the following three experiments, we compare the proposed method with these two methods, abbreviated as KLD_EW and UWT_Kmeans respectively.

4.4.2.2 First Experiment



(a) (b)

Figure 4.19: ROC curve of change detection using Kullback-Leibler divergence (a) The best results of the first dataset by TKLD_GΓD_MoLC and KLD_EW. (b) ROC curve corresponding to scale one and a window size of 33×33 pixels using the first dataset. We select the optimal threshold corresponding to the point closest to $(0.0, 1.0)$ on the ROC curve.

Table 4.5: AUCs for the first dataset. For a fixed window size and a fixed number of scales, the best and the worst values are marked in red and green.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.9507	0.9507	0.9501	0.9433	0.9356	0.9281	0.9200	0.9128
	TKLD_GGD_SE	0.9680	0.9680	0.9725	0.9752	0.9762	0.9771	0.9778	0.9788
	TKLD_GΓD_SE	0.9500	0.9500	0.9545	0.9547	0.9540	0.9550	0.9530	0.9537
	TKLD_GΓD_MoLC	0.9724	0.9724	0.9782	0.9834	0.9859	0.9876	0.9885	0.9893
2	TKLD_GGD_MLE	0.9487	0.9487	0.9474	0.9386	0.9295	0.9212	0.9117	0.9033
	TKLD_GGD_SE	0.9659	0.9659	0.9702	0.9726	0.9736	0.9744	0.9749	0.9755
	TKLD_GΓD_SE	0.9476	0.9476	0.9509	0.9506	0.9503	0.9511	0.9503	0.9504
	TKLD_GΓD_MoLC	0.9681	0.9681	0.9744	0.9798	0.9830	0.9851	0.9863	0.9872
3	TKLD_GGD_MLE	0.9454	0.9454	0.9443	0.9345	0.9249	0.9161	0.9060	0.8972
	TKLD_GGD_SE	0.9635	0.9635	0.9680	0.9702	0.9711	0.9719	0.9723	0.9729
	TKLD_GΓD_SE	0.9433	0.9433	0.9471	0.9462	0.9462	0.9464	0.9456	0.9454
	TKLD_GΓD_MoLC	0.9653	0.9653	0.9718	0.9770	0.9805	0.9829	0.9842	0.9852
KLD_EW		0.9729	0.9728	0.9692	0.9577	0.9444	0.9301	0.9164	0.9058

The AUCs using the first dataset are shown in Table 4.5. At each scale and with each window size, the best and the worst accuracy are marked respectively in red and green. It can be seen

clearly that GΓD estimated by MoLC is always the best method for any scale and any window size. On the contrary, GΓD estimated by the shape equation performs worst when the window size is smaller than 13×13; GGD estimated by maximum likelihood is always the worst method when the window size is larger than 13×13. As the window size increases, the accuracy of GΓD always increases. It is worth to note that the accuracy of GΓD estimated by the shape equation is always worse than GGD estimated by the shape equation and that GGD estimated by the shape equation is always better than maximum likelihood. From this, we conclude that a precise model does not always perform better than an inferior model, which depends much more on the estimation method. In addition, the shape equation is a better choice for estimating the parameters of GGD than maximum likelihood. For a fixed window size and method, the accuracy decreases as the number of scales increases. As AUC is a performance measure independent of thresholding, it reflects the general performance of the change detection method. To obtain a binary change map, the optimal threshold has to be selected. The TPRs and FARs are presented respectively in Table 4.6 and Table 4.7 based on the selected optimal thresholds. As can be seen, TPR and FAR are consistent with AUC. The best change map shown in Fig. 4.18(d) is obtained when the model GΓD is estimated using MoLC with a window size of 33×33 and one scale. Note that, to highlight the changes, the change map is shown using jet color map.

Table 4.6: TPR of the four methods on the first dataset.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.8936	0.8936	0.8907	0.8776	0.8676	0.8519	0.8386	0.8349
	TKLD_GGD_SE	0.9158	0.9158	0.9195	0.9275	0.9301	0.9408	0.9424	0.9518
	TKLD_GΓD_SE	0.8832	0.8832	0.8844	0.8848	0.8862	0.9033	0.9050	0.8984
	TKLD_GΓD_MoLC	0.9216	0.9216	0.9292	0.9359	0.9433	0.9537	0.9573	0.9640
2	TKLD_GGD_MLE	0.8920	0.8920	0.8841	0.8701	0.8533	0.8568	0.8305	0.8303
	TKLD_GGD_SE	0.9195	0.9195	0.9159	0.9148	0.9257	0.9333	0.9516	0.9527
	TKLD_GΓD_SE	0.8799	0.8799	0.8773	0.8752	0.8750	0.8980	0.9052	0.9052
	TKLD_GΓD_MoLC	0.9224	0.9224	0.9185	0.9330	0.9351	0.9448	0.9515	0.9585
3	TKLD_GGD_MLE	0.8858	0.8858	0.8807	0.8631	0.8474	0.8474	0.8326	0.8102
	TKLD_GGD_SE	0.9019	0.9019	0.9095	0.9086	0.9198	0.9389	0.9486	0.9424
	TKLD_GΓD_SE	0.8709	0.8709	0.8686	0.8699	0.8774	0.8876	0.9048	0.9116
	TKLD_GΓD_MoLC	0.9139	0.9139	0.9140	0.9271	0.9352	0.9402	0.9473	0.9557
KLD_EW		0.9186	0.9186	0.9182	0.8976	0.8839	0.8628	0.8551	0.8407

For the sake of comparison, the accuracy, TPRs and FARs of KLD_EW are also reported in the following tables. From the last row in Table 4.5, we can see that the accuracy of KLD_EW is quite high when the window size is smaller than 13×13. However, as the window size increases, its accuracy decreases dramatically. When the window size is larger than 17×17, it is worse than all other three methods, except for TKLD_GGD_MLE. The ROC curve of the best results obtained by TKLD_GΓD_MoLC and KLD_EW are shown in Fig. 4.19(a). In general, TPR and FAR follow the same observation. This can be explained by the fact that texture can be better characterized in the wavelet domain than in the spatial domain.

To compare with UWT_Kmeans, TPR and FAR are computed with different scales, which are $(0.4181, 0.2820), (0.6883, 0.2996), (0.7044, 0.2789)$. A two scale decomposition yields the best accuracies. However, they are lower than all other proposed methods. The reason could be that SAR images are degraded by speckle noise. Furthermore, k-means is an unsupervised approach, which degrades the accuracy significantly in the presence of outliers.

Table 4.7: FAR of the four methods on the first dataset.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.0983	0.0983	0.1085	0.1190	0.1322	0.1390	0.1507	0.1740
	TKLD_GGD_SE	0.0915	0.0915	0.0930	0.0978	0.0951	0.0932	0.0865	0.0867
	TKLD_GΓD_SE	0.0970	0.0970	0.1031	0.1170	0.1206	0.1342	0.1304	0.1200
	TKLD_GΓD_MoLC	0.0773	0.0773	0.0727	0.0702	0.0682	0.0696	0.0645	0.0625
2	TKLD_GGD_MLE	0.1013	0.1013	0.1059	0.1180	0.1295	0.1546	0.1549	0.1831
	TKLD_GGD_SE	0.0982	0.0982	0.0951	0.0949	0.0973	0.0948	0.0989	0.0950
	TKLD_GΓD_SE	0.1001	0.1001	0.1048	0.1192	0.1242	0.1388	0.1381	0.1306
	TKLD_GΓD_MoLC	0.0897	0.0897	0.0749	0.0770	0.0697	0.0728	0.0690	0.0678
3	TKLD_GGD_MLE	0.0996	0.0996	0.1096	0.1216	0.1331	0.1554	0.1670	0.1711
	TKLD_GGD_SE	0.0879	0.0879	0.0933	0.0946	0.0975	0.1041	0.1037	0.0953
	TKLD_GΓD_SE	0.1018	0.1018	0.1032	0.1263	0.1363	0.1437	0.1473	0.1452
	TKLD_GΓD_MoLC	0.0904	0.0904	0.0788	0.0792	0.0780	0.0751	0.0746	0.0743
KLD_EW		0.0781	0.0781	0.0867	0.1003	0.1245	0.1323	0.1497	0.1658

Table 4.8: The AUCs on the second dataset. For a fixed window size and a fixed number of scales, the best and the worst are marked in red and green.

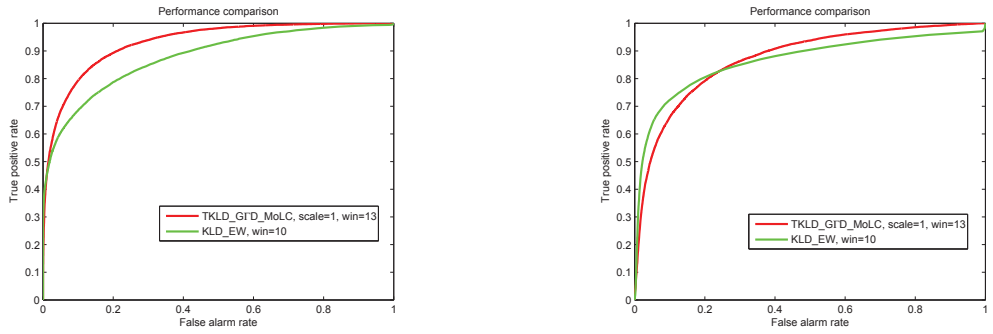
Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.8550	0.8550	0.8483	0.8218	0.7902	0.7603	0.7381	0.7246
	TKLD_GGD_SE	0.8777	0.8777	0.8807	0.8713	0.8514	0.8286	0.8080	0.7960
	TKLD_GΓD_SE	0.8653	0.8653	0.8700	0.8636	0.8514	0.8332	0.8164	0.8042
	TKLD_GΓD_MoLC	0.9232	0.9232	0.9314	0.9274	0.9110	0.8903	0.8722	0.8592
2	TKLD_GGD_MLE	0.8527	0.8527	0.8456	0.8161	0.7820	0.7469	0.7217	0.7054
	TKLD_GGD_SE	0.8733	0.8733	0.8766	0.8642	0.8409	0.8128	0.7899	0.7761
	TKLD_GΓD_SE	0.8613	0.8613	0.8646	0.8553	0.8384	0.8163	0.7977	0.7852
	TKLD_GΓD_MoLC	0.9117	0.9117	0.9214	0.9160	0.8961	0.8693	0.8476	0.8331
3	TKLD_GGD_MLE	0.8452	0.8452	0.8382	0.8069	0.7727	0.7362	0.7098	0.6920
	TKLD_GGD_SE	0.8653	0.8653	0.8684	0.8531	0.8278	0.7973	0.7735	0.7583
	TKLD_GΓD_SE	0.8528	0.8528	0.8555	0.8430	0.8236	0.7990	0.7795	0.7660
	TKLD_GΓD_MoLC	0.9014	0.9014	0.9107	0.9025	0.8803	0.8502	0.8270	0.8111
KLD_EW		0.8785	0.8785	0.8526	0.8016	0.7572	0.7241	0.7019	0.6879

4.4.2.3 Second Experiment

To further evaluate the observations, a second experiment was performed. The attained accuracies are presented in Table 4.8. From the results, the efficiency of TKLD_GΓD_MoLC in change detection is confirmed. For any scale and any window size, GΓD estimated by MoLC is always the best method followed by GGD estimated by shape equation; GGD estimated by maximum likelihood is always the worst alternative. In this evaluation, it can be observed that the accuracies of all methods reach their peak with a window size of 13×13. In addition, the observation that the accuracy decreases as the number of scales increases is also confirmed. The TPRs and FARs shown in Table 4.12 and Table 4.10 are consistent with AUC. The best results are obtained with a window size of 13×13 and a one scale decomposition; the corresponding change map is shown in Fig. 4.18(h).

The accuracy of KLD_EW ranks second with a window size smaller than 13×13. As in the first experiment, the accuracy decreases when the window size increases. The ROC curve of the best results obtained by TKLD_GΓD_MoLC and KLD_EW are shown in Fig. 4.20(a). It seems that KLD_EW has a better performance when the window size is small. In this test, the TPR and FAR of UWT_Kmeans are (0.4683, 0.2420), (0.5123, 0.2636), (0.4318, 0.2545), which are

much lower than the results by other methods. The accuracy of a two scale decomposition is also the best. All observations from the first experiment are confirmed in this evaluation.



(a) (b)

Figure 4.20: The best results of the second (a) and the third (b) data set by TKLD_GΓD_MoLC and KLD_EW.

Table 4.9: TPR of the four methods on the second dataset.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.7753	0.7753	0.7629	0.7390	0.7010	0.6661	0.6493	0.6403
	TKLD_GGD_SE	0.7940	0.7940	0.8102	0.7944	0.7778	0.7402	0.7153	0.7019
	TKLD_GΓD_SE	0.7800	0.7800	0.7884	0.7694	0.7818	0.7399	0.7089	0.6975
	TKLD_GΓD_MoLC	0.8472	0.8472	0.8476	0.8569	0.8243	0.7823	0.7640	0.7495
2	TKLD_GGD_MLE	0.7617	0.7617	0.7583	0.7330	0.7139	0.6642	0.6399	0.6179
	TKLD_GGD_SE	0.7882	0.7882	0.7836	0.7930	0.7654	0.7191	0.7017	0.6902
	TKLD_GΓD_SE	0.7628	0.7628	0.7721	0.7674	0.7543	0.7301	0.6967	0.6710
	TKLD_GΓD_MoLC	0.8220	0.8220	0.8349	0.8434	0.8098	0.7652	0.7389	0.7308
3	TKLD_GGD_MLE	0.7461	0.7461	0.7527	0.7215	0.6854	0.6543	0.6211	0.6006
	TKLD_GGD_SE	0.7836	0.7836	0.7939	0.7767	0.7459	0.6942	0.6746	0.6640
	TKLD_GΓD_SE	0.7659	0.7659	0.7678	0.7776	0.7374	0.6934	0.6749	0.6687
	TKLD_GΓD_MoLC	0.8082	0.8082	0.8211	0.8195	0.7854	0.7497	0.7247	0.7120
KLD_EW		0.7829	0.7829	0.7566	0.7104	0.6669	0.6431	0.7015	0.6593

4.4.2.4 Third Experiment

The accuracies of this evaluation using the third dataset shown in Fig. 4.18 are presented in Table 4.11. In line with the first two experiments, TKLD_GGD_MLE has the worst accuracy. However, in contrast to the first two experiments, TKLD_GGD_SE always performs best for all scales and window sizes, and is followed by TKLD_GΓD_MoLC. From this surprising result, we see that the accuracy depends more on the estimation method, although models are also important. Applying parameter estimation method based on the shape equation for GGD improves our results significantly compared with maximum likelihood estimation. When we compare these results with the first two experiments, where TKLD_GΓD_MoLC is always the best choice, we conclude that an inferior model with an efficient estimator can achieve a high accuracy. We can see from Table 4.11 that the accuracy of TKLD_GGD_SE reaches its peak when the window size is 25×25; the corresponding change map is shown in Fig. 4.18(l). In line with the first two experiments, the accuracy of KLD_EW decreases when the window size increases. Therefore, we

Table 4.10: FAR of the four methods on the second dataset.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.2282	0.2282	0.2276	0.2466	0.2734	0.2837	0.3038	0.3146
	TKLD_GGD_SE	0.2118	0.2118	0.2234	0.2192	0.2462	0.2573	0.2745	0.2880
	TKLD_GΓD_SE	0.2127	0.2127	0.2127	0.2023	0.2419	0.2422	0.2508	0.2682
	TKLD_GΓD_MoLC	0.1609	0.1608	0.1424	0.1521	0.1602	0.1637	0.1806	0.1976
2	TKLD_GGD_MLE	0.2197	0.2197	0.2265	0.2496	0.2932	0.2994	0.3160	0.3200
	TKLD_GGD_SE	0.2163	0.2163	0.2076	0.2273	0.2522	0.2612	0.2936	0.3024
	TKLD_GΓD_SE	0.2064	0.2064	0.2090	0.2137	0.2399	0.2628	0.2779	0.2722
	TKLD_GΓD_MoLC	0.1628	0.1628	0.1537	0.1610	0.1775	0.1873	0.2050	0.2273
3	TKLD_GGD_MLE	0.2139	0.2139	0.2337	0.2527	0.2759	0.3022	0.3138	0.3255
	TKLD_GGD_SE	0.2263	0.2263	0.2309	0.2304	0.2566	0.2595	0.2892	0.3040
	TKLD_GΓD_SE	0.2272	0.2272	0.2219	0.2490	0.2522	0.2538	0.2815	0.3010
	TKLD_GΓD_MoLC	0.1698	0.1698	0.1643	0.1700	0.1892	0.2129	0.2328	0.2551
KLD_EW		0.1951	0.1951	0.2142	0.2710	0.2944	0.3200	0.4000	0.3909

can reasonably conclude that KLD_EW performs better when the window size is small because in that case texture becomes less important for feature description. The best results obtained by TKLD_GGD_SE and KLD_EW are shown in Fig. 4.20(b). The true positive rate is completely consistent with the accuracy; however, the false alarm rate behavior differs slightly. When the window size is smaller than 13×13, TKLD_GΓD_SE is the worst option. For most window sizes, TKLD_GGD_MLE performs worst. The true positive rate and false alarm rate for a three scale decomposition of UWT_Kmeans are (0.3208, 0.2352), (0.4959, 0.2593), (0.4402, 0.2455), which are quite low.

Table 4.11: The AUCs on the third dataset. For a fixed window size and a fixed number of scales, the best and worst cases are marked in red and green respectively.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.8042	0.8042	0.8046	0.7957	0.7883	0.7741	0.7526	0.7351
	TKLD_GGD_SE	0.8597	0.8597	0.8709	0.8785	0.8856	0.8860	0.8821	0.8801
	TKLD_GΓD_SE	0.8130	0.8130	0.8285	0.8423	0.8569	0.8614	0.8601	0.8614
	TKLD_GΓD_MoLC	0.8372	0.8372	0.8504	0.8622	0.8696	0.8726	0.8721	0.8712
2	TKLD_GGD_MLE	0.7990	0.7990	0.7977	0.7867	0.7790	0.7620	0.7409	0.7232
	TKLD_GGD_SE	0.8515	0.8515	0.8619	0.8684	0.8758	0.8743	0.8704	0.8683
	TKLD_GΓD_SE	0.8058	0.8058	0.8207	0.8341	0.8466	0.8490	0.8474	0.8481
	TKLD_GΓD_MoLC	0.8255	0.8255	0.8393	0.8485	0.8556	0.8565	0.8538	0.8522
3	TKLD_GGD_MLE	0.7909	0.7909	0.7892	0.7787	0.7702	0.7537	0.7333	0.7162
	TKLD_GGD_SE	0.8418	0.8418	0.8516	0.8572	0.8639	0.8628	0.8589	0.8568
	TKLD_GΓD_SE	0.7937	0.7937	0.8080	0.8222	0.8338	0.8367	0.8347	0.8350
	TKLD_GΓD_MoLC	0.8156	0.8156	0.8285	0.8355	0.8418	0.8417	0.8381	0.8351
KLD_EW		0.8645	0.8645	0.8472	0.8098	0.7694	0.7316	0.6999	0.6793

From the three experiments, we have shown the most promising results for change detection based on a statistical modeling of wavelet sub-bands. In general, GΓD performs slightly better than GGD if an accurate estimation method is available. If the estimation method is not efficient enough, the accuracy obtained by an inferior model and a more accurate estimator can be better than an accurate model.

Table 4.12: TPR of the four methods on the third dataset.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.6997	0.6997	0.7094	0.7062	0.6952	0.6779	0.6643	0.6404
	TKLD_GGD_SE	0.7779	0.7779	0.7749	0.7913	0.8008	0.8030	0.8019	0.8002
	TKLD_GΓD_SE	0.7418	0.7418	0.7270	0.7548	0.7743	0.7842	0.7900	0.7834
	TKLD_GΓD_MoLC	0.7517	0.7517	0.7709	0.7749	0.7905	0.7854	0.7869	0.7894
2	TKLD_GGD_MLE	0.6978	0.6978	0.7039	0.6849	0.6907	0.6676	0.6615	0.6521
	TKLD_GGD_SE	0.7619	0.7619	0.7656	0.7767	0.7921	0.7896	0.7910	0.7876
	TKLD_GΓD_SE	0.7190	0.7190	0.7401	0.7396	0.7763	0.7852	0.7833	0.7785
	TKLD_GΓD_MoLC	0.7490	0.7490	0.7550	0.7648	0.7684	0.7697	0.7635	0.7506
3	TKLD_GGD_MLE	0.7014	0.7014	0.6921	0.6826	0.6708	0.6583	0.6433	0.6451
	TKLD_GGD_SE	0.7387	0.7387	0.7633	0.7556	0.7683	0.7840	0.7789	0.7691
	TKLD_GΓD_SE	0.7093	0.7093	0.7192	0.7440	0.7574	0.7534	0.7543	0.7642
	TKLD_GΓD_MoLC	0.7400	0.7400	0.7490	0.7548	0.7549	0.7618	0.7512	0.7385
KLD_EW		0.7771	0.7771	0.7631	0.7214	0.6849	0.6462	0.6172	0.6066

Table 4.13: FAR of the four methods on the third dataset.

Scale	Method	10	11	13	17	21	25	29	33
1	TKLD_GGD_MLE	0.2206	0.2206	0.2342	0.2535	0.2617	0.2761	0.3066	0.3187
	TKLD_GGD_SE	0.1973	0.1973	0.1721	0.1802	0.1807	0.1781	0.1855	0.1881
	TKLD_GΓD_SE	0.2504	0.2504	0.2085	0.2126	0.2152	0.2122	0.2162	0.2048
	TKLD_GΓD_MoLC	0.2096	0.2092	0.2045	0.1894	0.1993	0.1928	0.2025	0.2035
2	TKLD_GGD_MLE	0.2301	0.2301	0.2440	0.2487	0.2779	0.2876	0.3269	0.3519
	TKLD_GGD_SE	0.2014	0.2014	0.1859	0.1922	0.1973	0.1906	0.2017	0.2079
	TKLD_GΓD_SE	0.2496	0.2496	0.2438	0.2233	0.2440	0.2409	0.2378	0.2343
	TKLD_GΓD_MoLC	0.2340	0.2340	0.2134	0.2141	0.2121	0.2093	0.2199	0.2108
3	TKLD_GGD_MLE	0.2565	0.2565	0.2506	0.2641	0.2750	0.2936	0.3188	0.3496
	TKLD_GGD_SE	0.1993	0.1993	0.2099	0.1940	0.2005	0.2133	0.2187	0.2190
	TKLD_GΓD_SE	0.2644	0.2644	0.2508	0.2539	0.2523	0.2382	0.2406	0.2495
	TKLD_GΓD_MoLC	0.2489	0.2489	0.2334	0.2300	0.2254	0.2324	0.2385	0.2357
KLD_EW		0.1583	0.1583	0.1734	0.1947	0.2261	0.2422	0.2499	0.2672

4.4.3 Summary

In this section, an unsupervised method for SAR change detection in the wavelet domain based on statistical wavelet sub-band modeling is proposed and compared with other methods. A wavelet transformation is applied to decompose a given image into multiple scales. We assume that the wavelet coefficient magnitudes of each sub-band follow a Generalized Gaussian Distribution (GGD) or a Generalized Gamma Distribution (GΓD). Different parameter estimation methods have been investigated. A closed-form expression of the Kullback-Leibler divergence between two corresponding sub-bands from the same scale is computed and used to generate a final change map. This approach is comprehensively evaluated using different parameter settings, different scales, window sizes and estimators.

Through this study, SAR change detection in the wavelet domain shows promising results as texture can be better characterized in the wavelet domain than in the spatial domain. Fast parameter estimation methods for GGD and GΓD are investigated and applied. GGD has been applied in the wavelet domain as a standard model for coefficient modeling. As an alternative, GΓD has been recognized to be better than GGD due to the variability of image content. We conclude that the estimation accuracy depends heavily on the estimation method although models are important, too. Specifically, the maximum likelihood estimation for GGD always has the

lowest accuracy. On the contrary, an estimation method based on the shape equation for GGD improves our results significantly. In most cases, the estimation of GΓD by MoLC performs a little better than GGD in addition to its low computational complexity.

Although the methods can achieve good performance, there is still space for improvement. One possibility is to consider the inter- and intra-dependence of the wavelet coefficients. In this dissertation, we assume that the sub-bands are independent. Actually, sub-bands are dependent. The incorporation of a dependence term in the change map can improve the attainable accuracy.

Chapter 5

Spatial and Temporal High Resolution SAR Feature Extraction

High resolution SAR image feature extraction is not only an important part of image information mining, but is also quite challenging. Conventional texture features become insufficient for the description of high resolution SAR images, especially for urban areas. In this chapter, we first present a new perspective for feature extraction in section 5.1. Based on the intrinsic properties of high resolution SAR images, in section 5.2 we propose a new feature extraction method for the structure description of high resolution SAR images, which is inspired by the ratio edge detector Touzi *et al.* [1988]. Ratios in various directions within a local neighborhood are applied to enhance the Bag-of-Words (BoW) feature vectors that are generated using only the local image pixel statistics Cui *et al.* [2013a]. As ratios in the horizontal and vertical image directions can be considered as an extension of image gradients, they are used to adapt a Weber Local Descriptor (WLD) to SAR images. In section 5.3, we recall BoW feature extraction, followed by three contributions to the BoW feature extraction methodology in section 5.4. First, we propose to use the pixel brightness values within a local neighborhood as low level features for the BoW feature extraction. Furthermore, we propose to use a random dictionary instead of unsupervised clustering for dictionary learning. In parallel, we develop a new feature coding method, namely incremental coding. We demonstrate that our methods can achieve a significantly higher accuracy than state-of-the-art methods for SAR image classification. In section 5.5, parameters being needed for the BoW model are evaluated rigorously, like the sampling strategy, or the choice of patch size and dictionary size. In the last section 5.6, we extend the BoW model to SAR image time series, where the feature extraction method is called Bag-of-Spatial-Temporal-Words (BoSTW), which performs better than a concatenation of other conventional features.

5.1 High Resolution SAR Image Feature Extraction

5.1.1 A New Perspective for High Resolution SAR Image Feature Extraction

Nearly all SAR texture features available in the literature only focus on the texture aspect, which performs quite well for target areas with fully developed speckle. However, due to the advent of high resolution SAR instruments, like TerraSAR-X, TanDEM-X, COSMO-SkyMed, etc., the structure in a local context becomes important for efficient image classification and recognition, as shown in Fig. 5.1. This poses challenges in developing new features for better discrimination. Although texture features, such as GLCM Haralick *et al.* [1973], Gabor filters Manjunath & Ma [1996], and GMRF Clausi & Yue [2004], have played an important role in

low and medium resolution SAR image interpretation, the complex structure in a local context is becoming more critical for Very High Resolution (VHR) SAR image characterization than texture features Popescu *et al.* [2012], which is demonstrated in Fig. 5.5. Obviously, without prior knowledge and consideration of the complex structure arrangement in the local context, a scene classification cannot be efficiently achieved. In this sense, only a group of pixels can have a clear meaning and can be interpreted appropriately. The image content becomes more diverse. Therefore, we propose a higher patch level analysis to include more information for SAR image analysis. In the literature, several works have been done for structure description. A Multilevel Local Pattern Histogram (MLPH) was proposed by Dai *et al.* [2011], which shows a better performance than GLCM, Gabor and GMRF. MLPH represents the size distribution of local patterns in an image. A local pattern is a bright or dark region which can be derived by thresholding. In Popescu *et al.* [2012], a patch level contextual descriptor of SAR images based on a Fourier transformation was proposed and compared with GLCM and has shown a superior classification performance.

To the best of our knowledge, the first work on patch level analysis in the remote sensing community was carried out by Shyu *et al.* [2007]. Based on this motivation, we have developed two contributions to feature extraction for high resolution SAR image content description. The first one is a new feature extraction method for the structure description of high resolution SAR images, which is inspired by the ratio edge detector. Ratios in various directions within a local neighborhood are applied to enhance the BoW feature vectors that are generated using only the local image pixel statistics. As the ratios in the horizontal and vertical image directions can be considered as an extension of image gradients, they are used to adapt the WLD to SAR images as WLD proposed by Chen *et al.* [2010] is a joint histogram of local contrast and gradients. The second contribution that we propose is to use the pixel brightness values within a local neighborhood as low level features in the BoW model. Without any additional feature extraction, we demonstrate that pixel brightness values within a compact local neighborhood give a surprisingly better performance than other state-of-the-art texture features. Furthermore, we develop a new feature coding method called incremental coding, which learns the feature coding of an image based on the other images from the same class which have already been encoded by the algorithm. We have demonstrated that incremental feature coding gives an accuracy of almost one hundred percent and performs significantly better than state-of-the-art feature encoding methods for SAR image classification. In addition, the parameters involved in the BoW model are evaluated rigorously, like the sampling strategy, the patch size and the dictionary size. In the last part of this chapter, we extend the BoW model to SAR image time series, where the feature extraction method is called BoSTW, which performs better than a concatenation of other conventional features.

5.2 Features Based on the Ratio Detector

5.2.1 Ratio Detector

Due to the presence of multiplicative speckle noise, conventional differential-based edge detectors are insufficient for SAR images. To combat against speckle, a ratio edge detector was proposed in Touzi *et al.* [1988]. The ratio detector is defined as the ratio of the means of two neighborhoods on the opposite sides of a pixel. To detect all possible edges, the ratio detector has to be applied in all possible directions. The edge response of a pixel is the maximum of all edge responses in all directions. For computational efficiency, a local window centered at a pixel is split into two contiguous neighborhoods. Four principal directions $i = 0, ..., 3$, etc., 0°, 45°, 90°, 135°

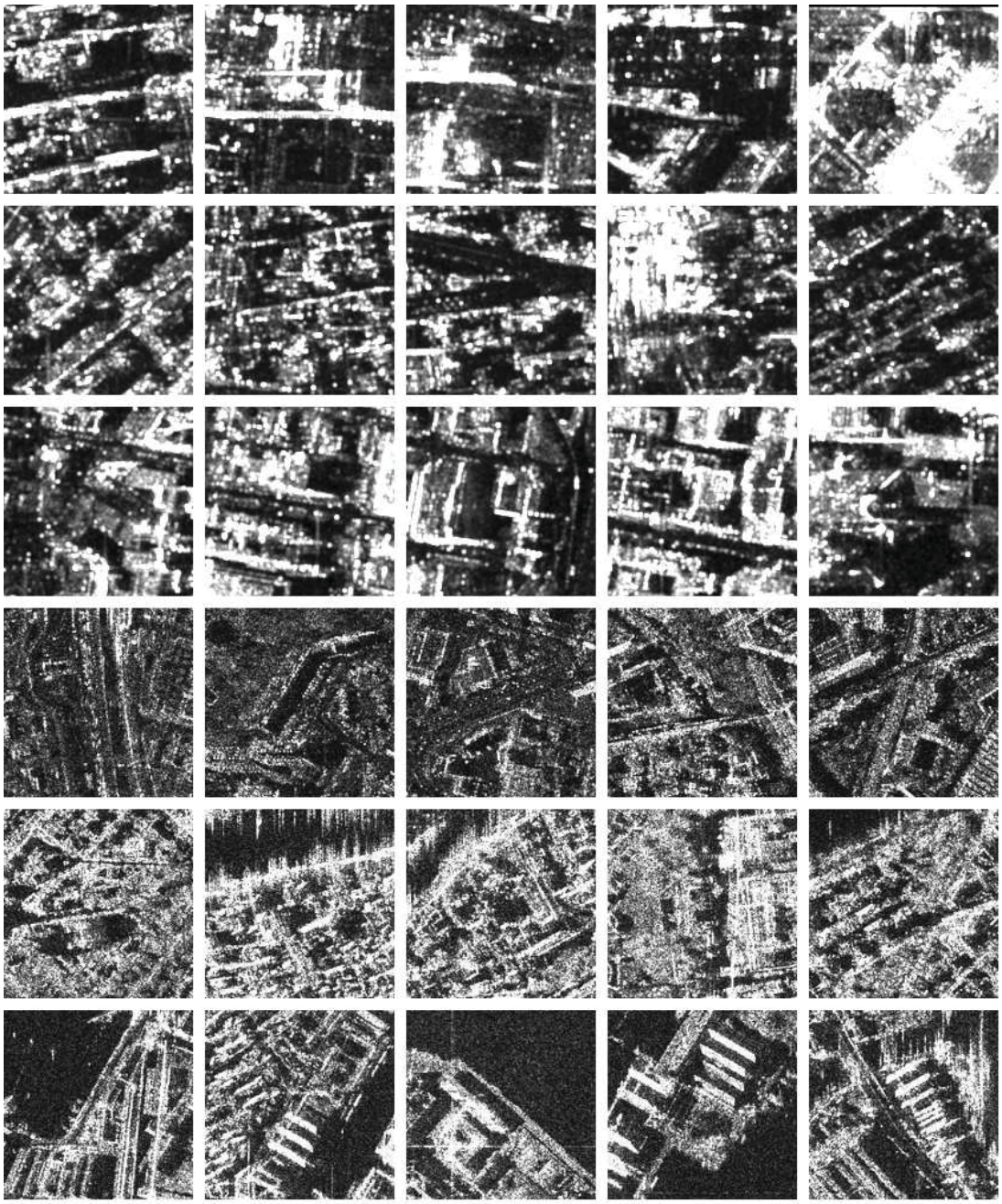
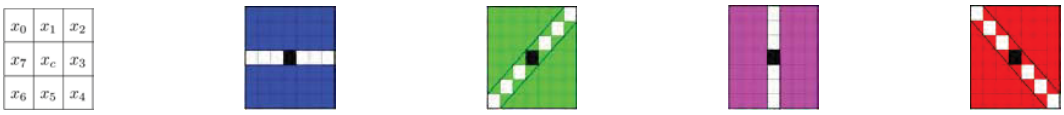


Figure 5.1: Examples of structural patches.

are assumed in Fig. 5.2 and the ratios of local means $(\mu_1^i,\ \mu_2^i)$ of the two neighborhoods are computed.

The edge responses of the four principal directions are defined as

$$r^i = 1 - \min\Big(\frac{\mu_1^i}{\mu_2^i}, \frac{\mu_2^i}{\mu_1^i}\Big) \quad i = 0, ..., 3. \tag{5.1}$$



(a) (b) (c) (d) (e)

Figure 5.2: 8-neighborhood and four directions (0° , 45°, 90°, 135°) used for the definition of the SAR image ratio detector.

The final edge response of a pixel is given as $R = \max(r^i), i = 0, ..., 3$, which we call the mean ratio. The mean ratios of all assumed directions are used as low level features to improve the BoW method in the next section.

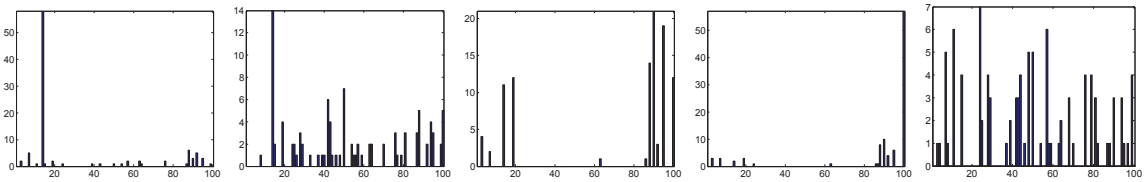
Based on the ratio detector, the Ratio of Mean Differences (RMD) is defined as

$$R_{SAR} = \frac{D_v}{D_h} = \frac{\mu_1^1 - \mu_2^1}{\mu_1^3 - \mu_2^3}, \tag{5.2}$$

where $D_v = \mu_1^1 - \mu_2^1$ and $D_h = \mu_1^3 - \mu_2^3$ are the differences of the local means of the two neighborhoods corresponding to both vertical and horizontal directions, namely the two blue and magenta regions shown in Fig. 5.2(b) and (d). The RMD can be considered as an adaptation of the gradients to SAR images. Note that the RMD is defined based on a local window. Therefore, we can define a multi-scale RMD by a series of windows with increasing size ($s = 3 \times 3, 5 \times 5, ...$). The RMD is used as an alternative to gradients in the adaptation of the WLD method to SAR images in the next section.

5.2.2 Incorporation into Bag-of-Words Method

Borrowed from the text analysis community and texton analysis, the BoW model has quickly become a standard state-of-art method for visual classification in the computer vision community. In the BoW model, an image is represented as a collection of local features, key points detected by SIFT, densely sampled patches, etc. Local features are extracted from the local neighborhood around SIFT points or densely sampled patches. To construct the vocabulary, a clustering, usually k-means, is applied to find clusters. The cluster centers are used as a vocabulary for computing word occurrence histograms. After vocabulary generation, each local feature vector is assigned to the closest cluster; thus an image can be represented as a word occurrence histogram. A recent study by van Gemert *et al.* [2010] tries to assign a local feature vector to multiple words. For the sake of simplicity, nearest word assignment is employed in our case. A critical step in this framework is local feature extraction, which has attracted increasing interest in local descriptor development by the computer vision community. However, in the SAR community, almost no solution has been proposed to develop local SAR image descriptors, which is the main motivation of this chapter. We started from simple local statistics, i.e., mean and variance of image patches, which is abbreviated as BoW_MV in the following sections. For fully developed speckle, the relation between mean and variance given as $L = \frac{m^2}{\sigma}$ holds, where L is the number of looks, m is the mean and σ is the variance. However, for areas with strong structures, this relation does not hold any more. Therefore, we propose to incorporate both the mean and variance as separate parameters into the local feature vector. To describe the structure of a local patch, we add the mean ratios in different directions in addition to the local mean and variance, when we extract the local features. In this case, the feature extraction method is abbreviated as BoW_MVR in the following sections. The BoW_MVR features of 5 classes are shown in Fig. 5.3. As can be observed, the word histogram has strong discriminative capabilities for SAR image classification.

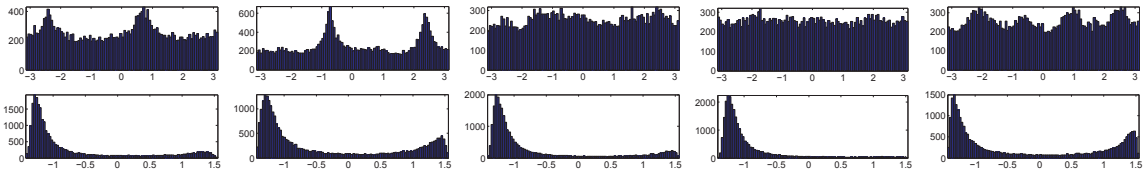


(a) (b) (c) (d) (e)

Figure 5.3: From left to right, word histograms of the classes (a) *bridge*, (b) *harbor*, (c) *alluvial deposits*, (d) *vegetation and forest*, and (e) *urban* after incorporating the mean ratios.

5.2.3 Weber Local Descriptor (WLD)

WLD was proposed by Chen *et al.* [2010] for optical image retrieval inspired by Weber's law, which states that the change of a stimulus that will be just noticeable is a constant fraction of the original stimulus. If the change is smaller than this constant ratio, it cannot be recognized. Based on this idea, WLD is proposed for texture characterization, which is composed of two components: differential excitation ξ and orientation θ.



(a) (b) (c) (d) (e)

Figure 5.4: From left to right, histograms of orientation and differential excitation (in the first and second row) respectively of the classes (a) *bridge*, (b) *harbor*, (c) *alluvial deposits*, (d) *vegetation and forest*, and (e) *urban*.

The differential excitation is defined as

$$\xi(x_c) = \arctan\Big[\sum_{i=0}^{n-1}\Big(\frac{x_i - x_c}{x_c}\Big)\Big], \tag{5.3}$$

where n is the number of neighboring pixels, which is 8 in the case of Fig. 5.2(a). The orientation is given as the gradient orientation, defined by

$$\theta(x_c) = \arctan2\Big(\frac{x_7 - x_3}{x_5 - x_1}\Big). \tag{5.4}$$

Based on the two terms, a joint histogram can be constructed, followed by converting it to a one-dimensional histogram, which is the WLD descriptor. The advantage of this feature is that it considers not only the local contrast, but also the structure information represented by the gradient histogram.

5.2.4 Adapted Weber Local Descriptor

Although Weber's law is a theory from the area of visual perception, its feature extraction principle is applicable to SAR images. However, in the case of SAR images, their discriminative ability for image indexing is decreased dramatically by multiplicative speckle noise. To combat

against speckle, we propose a solution to replace the gradient in WLD by the RMD, leading to the following orientation component:

$$\theta(x_c)_{SAR} = \arctan2(R_{SAR}) = \arctan2\Big(\frac{D_v}{D_h}\Big). \tag{5.5}$$

Furthermore, the differential excitation is adapted by

$$\xi(x_c)_{SAR} = \arctan\Big[\sum_{i=0}^{3}\sum_{j=1}^{2}\Big(\frac{\mu_j^i - x_c}{x_c}\Big)\Big]. \tag{5.6}$$

As an example, the histograms of the orientation and differential excitation for SAR images from 5 classes are shown in Fig. 5.4. As can be seen, the adapted WLD is discriminative for structure. Visually, the 5 classes can be well discriminated.

Based on the two adapted components $\theta(x_c)_{SAR}$ and $\xi(x_c)_{SAR}$, the WLD for SAR images can be defined as the joint histogram $H\Big(\xi(x_c)_{SAR}, \theta(x_c)_{SAR}\Big)$. Therefore, there are two adjustable parameters, the number of bins for excitation C and the orientation T. Following the same strategy as for WLD, this joint histogram is converted to a one-dimensional histogram. The adapted WLD includes not only local statistics, but also local structure information, resulting in an improved performance in SAR image indexing.

5.2.5 Evaluation and Discussion

To evaluate the performance of the two proposed methods (BoW_MVR and the adapted WLD), a SAR image database was prepared by tiling a TerraSAR-X image of 10881 × 15782 pixels covering the area of Venice (Italy). The data product is a Multilook Ground Detected (MGD), Spatially Enhanced (SE) high resolution Spotlight mode image with single polarization (HH). Its pixel spacing is 1.25 m and the incidence angle is 38°. The number of looks in range and azimuth are 3.08 and 2.60 respectively. This TerraSAR-X image was tiled into patches of 160×160 pixels covering an 200×200 m^2 on ground. From 1026 patches, 17 different classes were extracted. From these classes, we selected 14 classes shown in Fig. 5.5 with more than 10 patches having the same semantics.

All experiments conducted below are based on an active learning system for interactive SAR image indexing Cui *et al.* [2013b]. Two important components in our active learning system are two modules for classifier training using the already labeled images and sample selection which selects the most informative samples for manual labeling. These two components work alternatively, which can significantly reduce the human labeling effort and achieve a better performance for image indexing. The classifier we adopted is a C-SVM (See section 6.2); its kernel is a chi-square function that is well suited and efficient for histogram-based feature vectors. The only parameter in C-SVM is the penalty parameter C. This parameter is empirically set to 1000. During each iteration, samples close to the class boundary are selected from the unlabeled pool for subsequent manual labeling. These are the most important samples to train a classifier effectively and quickly. The accuracy measures we used for evaluation are precision and recall. Precision is the fraction of retrieved documents that are relevant to the search and recall is the fraction of the documents that are relevant to the query that are successfully retrieved.

In this section, we describe three experiments that have been carried out for further analysis. The first one is to demonstrate the advantages of BoW_MV over Gabor texutre and the improvements obtained by BoW_MVR, while the second experiment compares the adapted WLD with the original WLD. The last experiment compares these two methods with state-of-the-art SAR

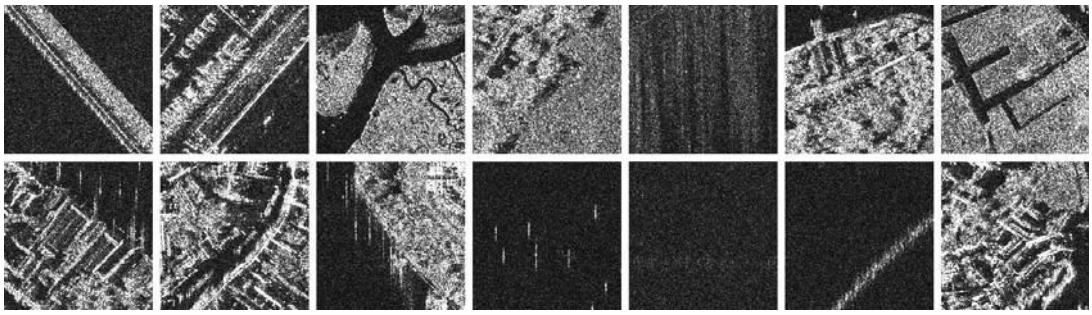


Figure 5.5: 14 selected classes and their semantic labels: *bridge, harbor, alluvial deposits, airport, breaking waves, vegetation and forest, agriculture, urban with water, urban, vegetation with water, buoy, water, water with boats*, and *vegetation with buildings*. The first row shows example images from class 1 to 7 while the second row contains example images from class 8 to 14. In VHR SAR images, detailed structures appear and become important for interpretation. Without consideration of the complex structure arrangement in their local context, scene classification cannot be efficiently achieved. Classical pixel based classification and segmentation methods cannot solve this issue. Therefore, we need an image patch-based analysis taking into account the local structures.

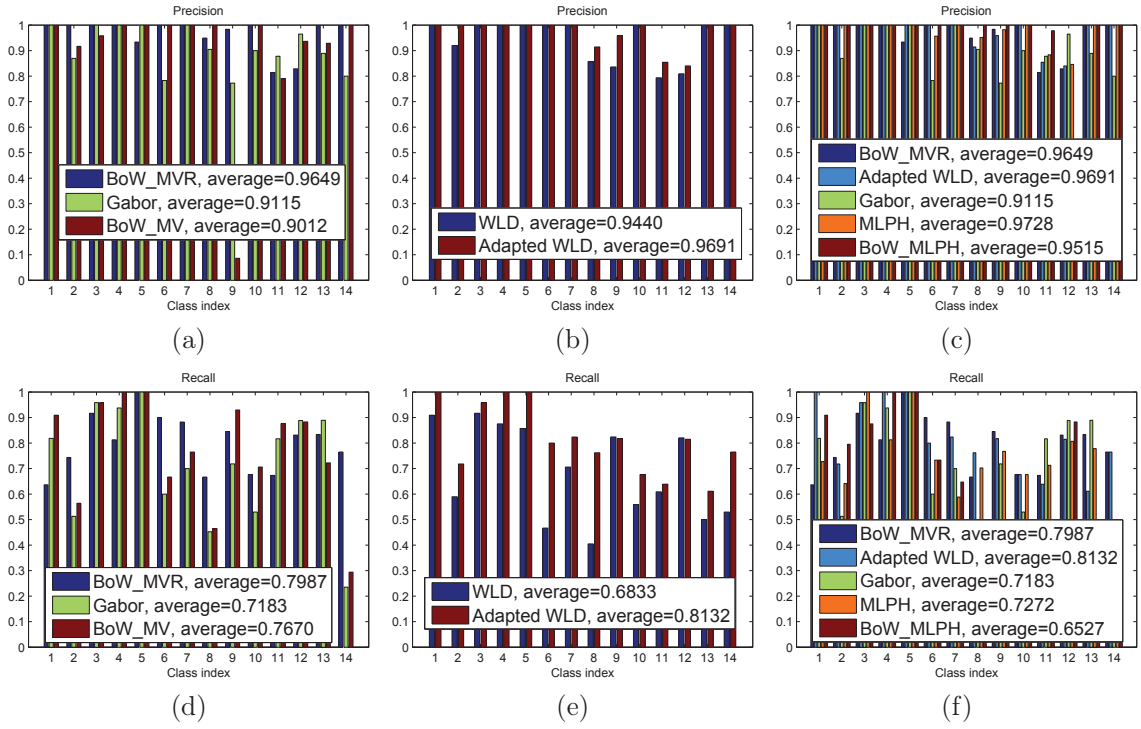


Figure 5.6: Performance comparison of different features for SAR image indexing: (a) (d) precision and recall comparison of BoW and Gabor texture; (b) (e) precision and recall comparison of two BoW models; (c) (f) precision and recall comparison of WLD and adapted WLD.

feature extractors. Furthermore, the BoW method using MLPH as low level features, abbreviated as BoW_MLPH, is also applied and compared because the BoW feature vector is a mid-level

feature, which can be generated using any low level features. The Gabor texture features used in all the three experiments are the mean and variance of each sub-band as proposed by Manjunath & Ma [1996].

For BoW classification, each image is tiled into patches with a size of 16×16 pixels, corresponding to a same size sliding window for local feature extraction. In the first test, we compare BoW_MV with Gabor texture features generated with 4 scales and 6 orientations. The clustering algorithm we adopted for vocabulary generation is the Enhanced Linde-Buzo-Gray (ELBG) algorithm by Patané & Russo [2001], which converges towards a better local minimum compared with k-means. The number of clusters we assumed is 100 because it has been demonstrated by Wang *et al.* [2011]; Xu *et al.* [2010] that there is no big improvement using a large vocabulary size. Each feature vector is assigned to its closest word in the feature space. The precision and recall results are shown in Fig. 5.6(a) and (d). A surprising result is that BoW_MV performs better than Gabor texture feature extraction with the exception of only two classes (12nd and 13rd). These two classes are respectively *pure water* and *water with boat*, which are quite homogenous as shown in Fig. 5.5. The average recall values of these two methods are 71.83% and 76.70%. A 5% improvement in recall is achieved using only the local means and variances, which is a quite promising improvement. The precision and recall results of both BoW_MVR and BoW_MV are also shown in Fig. 5.6(a) and (d). The performance of BoW_MVR is slightly better than BoW_MV. The average precision and recall results improve by 6.37% and 3.17% respectively. However, for well-structured urban classes, especially the 9th class, there is a large improvement in recall.

The second evaluation is conducted to compare the adapted WLD with the original WLD method, where the features are extracted from the whole patch, without tiling it into sub-patches. In both feature extraction methods, the involved parameters (i.e., the number of bins for orientation and excitation) are set to the same values, $C = 18$ and $T = 8$. The window sizes used for local feature extraction by the adapted WLD and the original WLD are 7×7 and 3×3 pixels respectively. A larger window size is needed for SAR images to alleviate the impact of speckle. The respective precision and recall results are shown in Fig. 5.6(b) and (e). It can be clearly observed that both precision and recall are significantly improved for all classes, which confirms the effectiveness and robustness of the adapted WLD for SAR images. The average improvements of precision and recall are 2.51% and 12.99% respectively. The improvement of the adapted WLD can be explained by the fact that SAR images are impacted significantly by multiplicative noise, which is not the case for optical images. The RMD alleviates the influence of multiplicative speckle by averaging over all pixels within a local neighborhood and taking the ratio of means. The effectiveness of RMD for structure description in SAR images is also confirmed by the improvement in accuracy.

The last experiment is carried out to compare our methods with state-of-the-art feature extractors, as well as the BoW method based on these features. The comparison of the state-of-the-art feature extractors was done in our previous work by Dumitru *et al.* [2012], where we compared GLCM, Short Time Fourier Transformation (STFT), Gabor textures and Quadrature Mirror Filter (QMF) texture features with different parameter settings. We concluded that Gabor texture features perform better than the other techniques. As shown in the first experiment, Gabor features are inferior to BoW_MV. Therefore, in the following, we do not consider GLCM, STFT, QMF, and Gabor features anymore and compare our results only with MLPH and the related BoW method called BoW_MLPH. In the parameter settings of MLPH, there are mainly two parameters which are the number of levels and the increase rate of the threshold. They are set to 5 and 3 respectively. In contrast to the original implementation by Dai *et al.* [2011], where a sliding window is placed around each image pixel to extract a local pattern histogram,

we use a constant patch size of 16×16 pixels,, which is equivalent to the sliding window in the original implementation. The precision and recall results of these four features are shown in Fig. 5.6(c) and (f). As a baseline for comparison, the precision and recall values of the Gabor method are also plotted. MLPH performs slightly better than Gabor. Similarly, BoW_MLPH is inferior to MLPH. From this result, we see that it is not always advantageous to apply the BoW method using a low level feature. In contrast, the adapted WLD and BoW_MVR achieve a comparable performance with similar precisions of 96.91% and 96.49% and recalls of 81.32% and 79.87%, because both of them consider not only the local statistics but also the local structure information in a spatial context.

To overcome the drawbacks of pixel level VHR SAR image analysis techniques, an image-patch-based method is proposed for SAR image interpretation. Inspired by the ratio edge detector, a new feature extraction method represented by the mean ratios in different directions is proposed for high resolution SAR image characterization. Based on the mean ratios, two simple yet powerful and robust feature extractors are proposed for SAR image patch indexing. The first one is the BoW model using not only the local statistics, i.e., the local means and variances, but also the mean ratios in different directions. The other one is an adaptation of the original WLD method to SAR images by substituting the gradients with the ratios of the mean differences in vertical and horizontal directions. These two features are evaluated and compared with state-of-the-art features based on an active learning system using a database consisting of one thousand TerraSAR-X images. Our evaluations and comparisons have demonstrated the effectiveness of the proposed feature extractors for structure description because both of them consider not only the local statistics, but also the local structure context. The adaptation of WLD shows significant improvement compared to WLD in SAR image indexing. In addition, the discrimination capabilities of the BoW model using only the mean and variance as low level features have been improved considerably after incorporating the mean ratio.

5.3 The Bag-of-Words Method

In this section, we focus on the general framework of the BoW model and review three important components: local feature extraction, codebook generation, and feature assignment.

5.3.1 Local Feature Extraction

The BoW assumption is that an image is considered an orderless collection of patches, which can be extracted by an interest point detector or can be densely sampled. Local features are extracted to represent the local appearance, which will be quantized or coded to generate a codebook.

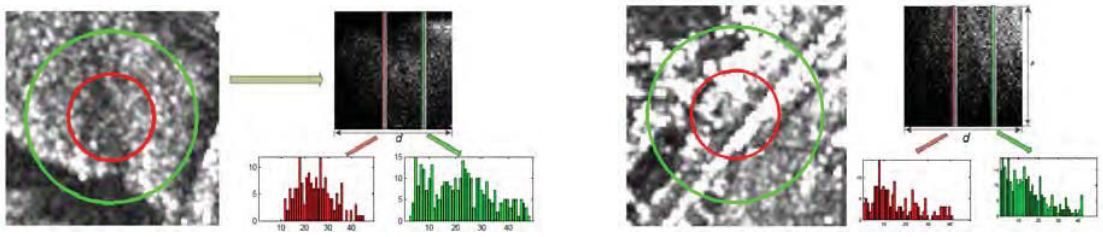
5.3.1.1 Sparse Feature Detection

There are two directions of research for feature detection. The first one evolved from image matching, with the aim of detecting features that are both discriminative and robust to minor geometrical and photometric transformations, like SIFT or SURF. The second one is based on the principle of visual saliency trying to detect salient points under some criterion. The first group of methods is mainly based on scale space theory, which represents an image at multiple resolutions through a convolution with a Gaussian filter with an increasing scale parameter. A typical and popular example is SIFT, which detects the extremal response points in the Difference-of-Gaussian space as interest points. The Difference-of-Gaussian space is generated

by successive subtraction of images at two close scales. Another important detector is the Harris-Affine Detector proposed by Mikolajczyk & Schmid [2004], which gives a multi-scale representation for the Harris interest point detector and then selects points at which a local measure (the Laplacian) is maximal over scales. This provides a set of distinctive points which are invariant to scale, rotation and translation as well as robust to illumination changes and limited changes of viewpoint. In addition to these two detectors, there are many others. A comprehensive evaluation was carried out by Mikolajczyk & Schmid [2005]. An example from the second group is Scale Saliency proposed by Kadir & Brady [2001], which detects salient regions with high entropy in the scale space. Similarly, an information theoretic approach to saliency detection was proposed by Bruce & Tsotsos [2009] based on the information theoretic formulation dubbed Attention based on Information Maximization (AIM). On the basis of this sparse feature detector, a lot of work in the framework of BoW has been done. However, a fundamental question is whether a detector provides enough distinctive features for image classification, because most of the feature detectors have been developed for geometrical image matching, although they can achieve good classification results on some occasions.

5.3.1.2 Dense Feature Extraction

There are works showing that dense sampling on a regular grid, or even random sampling performs much better than sparse feature detectors, such as Maree *et al.* [2005]; Nowak *et al.* [2006]. Intuitively, sparse feature detectors detect only the interest or salient points, thus the entire image will not be covered. In contrast, dense sampling gives a better coverage of the entire image and a constant amount of features per image. In this case, not only the interest points are detected but also other patches will be included, which might not be necessary for feature matching but they contain valuable information about the image content for image classification. In this section, we consider only dense sampling and random sampling. However, there are still two more important parameters in dense sampling, which are the *patch size* and the *overlap*. If the patch size is too large, the curse of dimensionality arises. Similarly, if the overlap is too large, we would have a huge amount of features, which need a large memory and increase the computation during the subsequent codebook generation phase. Selecting an appropriate patch size and overlap becomes important for practical applications. These two parameters will be evaluated in the following paragraphs of section 5.5.



(a) (b)

Figure 5.7: Two descriptors: (a) SPIN image; (b) RIFT.

Since the BoW method creates mid-level features using the underlying low level features, thus, the discriminative low level features should be carefully selected. In general, local rotation invariant features are preferable for image classification. As the local patches are usually small, in principle, the maximum number of features that we can extract is given by the image size. Then,

basic statistics can be extracted as low level features to generate the BoW feature vector. As we tested previously in section 5.2.5, local statistics, i.e., mean and variance, are already better than Gabor texture features, which shows the promising ability of the BoW method for SAR image classification. In this line of research, there are some interesting works on local feature extraction. Two local descriptors, the SPIN image and RIFT, were proposed by Lazebnik *et al.* [2005]. The SPIN descriptor is a joint histogram of the distances to the center pixel of a local patch and the corresponding intensities, as shown in Fig. 5.7(a). The RIFT descriptor is a joint histogram of distances and the gradient orientations in a local patch. To reach rotation invariance, each gradient orientation is computed with respect to the direction pointing to the pixels in each ring, as shown in Fig. 5.7(b). Following the texton theory, Varma & Zisserman [2005], as well as Varma & Zisserman [2009], proposed to use all the pixel values within a local patch instead of the filter responses for image classification. They claimed that compact local patches can achieve better results than texton histograms of filter responses. Based on this work, random projections were applied by Liu & Fieguth [2012] to reduce the dimension of the local feature vector and a significant improvement in classification accuracy and reductions in feature dimension were claimed. However, it was observed that a random projection of a local feature is not rotation invariant, thus sorted random projections of five local features (shown in Fig.5.8) were proposed later by Liu *et al.* [2011b, 2012], which was claimed to achieve a significant improvement compared to the random projection of a local patch. However, the validity of this technique has not been evaluated so far for SAR image classification. In addition, we think that the rotation invariance of the local features is not very important because the local feature extraction is an intermediate step and our final goal is to extract discriminative global features of the entire image to achieve better classification results. Even if a local feature is not rotation invariant, the final word frequency histograms can have almost the same performances in image classification as long as there are a sufficient number of well-separated local features in the feature space.

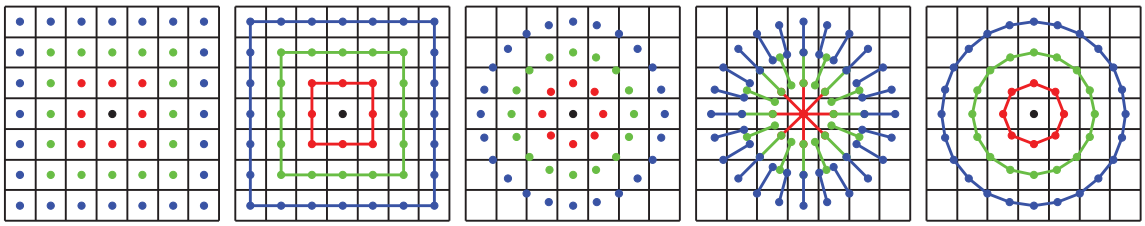


Figure 5.8: Five local descriptors of a 5×5 local neighborhood in the method of Sorted Random projection (SRP) proposed by Liu *et al.* [2011b, 2012]: (a) SRP Global: sort all the pixel values in the local neighborhood; (b) SRP Square: sort the pixel values in each surrounding box and concatenate them as a feature vector; (c) SRP Circular: sort the pixel values in each circular ring and concatenate them as a feature vector; (d) SRP Radial-diff: sort the differences of two corresponding pixels in two enclosing circular rings and concatenate them as a feature vector; (e) SRP Angular-diff: sort the differences of two enclosing pixels in each circular ring and concatenate them as a feature vector.

Based on our experience, we propose to use all pixel values in a very small local neighborhood, like 3×3 pixels, as low level features for codebook learning. The result is much better than many other low level features for SAR image classification, as conventional feature extraction methods lose an amount of information being contained in an image. In contrast, the pixel values comprise

the full information. This confirms the finding by Varma & Zisserman [2009].

5.3.2 Codebook Generation

A codebook is learned, usually in an unsupervised manner, to encode the appearance of local neighborhoods. A codebook consisting of cluster prototypes is a quantization of the continuous feature space. The conventional method to generate a codebook is to apply unsupervised learning methods to separate different groups of features in the feature space. The cluster prototypes are stacked to form a codebook. As an efficient unsupervised learning method, k-means clustering is usually employed by minimizing the within cluster sum of squared distances. In this procedure, normally the nearest neighbor principle is used based on Euclidean distance. Therefore, the initial centroids and the number of clusters are crucial parameters. Actually, this is a classical problem known as model selection. The number of clusters has a substantial influence on the classification accuracy and represents the dimension of the final feature vectors. On the other hand, the computational effort should be considered in the case of a large scale database. When dealing with a small database, k-means can be applied many times and the best corresponding codebook can be selected. In spite of this issue, some approximate clustering algorithms by Philbin *et al.* [2007] and Nister & Stewenius [2006], or other efficient methods, like random forest published by Moosmann *et al.* [2006] can be applied.

As an alternative to k-means clustering for codebook generation, GMM presented in section 4.1.6.1 can also be applied to generate a better codebook. In this method, each cluster is assumed to follow a normal distribution. Given a set of local descriptors, a Gaussian mixture model with k components is learned using the EM algorithm (See section 4.1.6.1), where $\theta = (\pi_1, \mu_1, \mathbf{\Sigma}_1, ..., \pi_K, \mu_K, \mathbf{\Sigma}_K)$ are the parameters governing the GMM. If we assume that the covariance matrix of each component is diagonal, there remain only $(2D+1)K$ parameters with K being the number of components and D being the dimension of the local descriptor. After learning the model, the posterior distribution of a local descriptor x_i , which is considered as the soft assignment to components, is

$$q_{ki} = \frac{p(x_i|\mu_k, \mathbf{\Sigma}_k)\pi_k}{\sum_{i=1}^{K} p(x_i|\mu_i, \mathbf{\Sigma}_i)\pi_i}, \quad k = 1, ..., K \tag{5.7}$$

5.3.3 Feature Assignment and Encoding

After learning the codebook, an encoding schema is applied to assign local descriptors to the elements in the codebook. Usually, each local descriptor is assigned to the nearest element in the codebook and the word frequency histogram for the entire image is employed as a feature vector representation of the image. However, this will be problematic if there are some local descriptors that are equally close to more than one cluster center. In this case, it is probably not wise to apply a nearest neighbor assignment. In addition, there are some code words that are more important than others for classification. The intuitive idea is to assign weights to the words when we compute the global representation of the entire image. This is the motivation of the popular Term Frequency-Inverse Document Frequency (TF-IDF), which reflects the relative importance of the words in the database, and is widely used in document classification. TF-IDF is defined as $h_i \log(\frac{N}{N_i})$, where h_i is the frequency percentage of the ith word, N is the total number of images in the database, and N_i is the number of images where the ith word occur. A weight is high when the word is frequent in an image but rare in the others. For those words that are quite common in most images, the corresponding weights are quite low. Therefore, the discriminative words of each class have high weights.

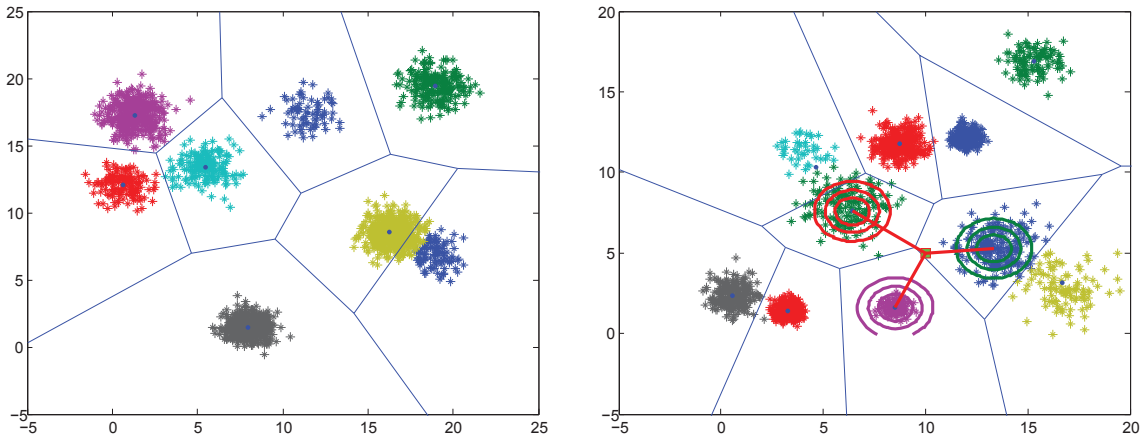


Figure 5.9: Codebook generation: (a) A codebook with 8 elements is learned using k-means, which are the cluster prototypes; (b) Kernel codebook, the $k = 3$ nearest clusters of the marked point are given different weights according to a Gaussian function centered at each cluster center. The weight assigned to each cluster represents the relative importance of each cluster to the current feature point. This idea is similar to Locality-constrained Linear Coding (LLC) Wang *et al.* [2010], where the weights are the local reconstruction coefficients of a least squares problem, rather than the local probabilities.

Soft assignment has also been developed to reduce the information loss in vector quantization. Soft assignment is to assign a feature vector to multiple clusters with different weights. As an example, a kernel codebook was proposed by van Gemert *et al.* [2010], which assigns high weights to the closer clusters and small weights to the far ones. A Gaussian function $K(\mathbf{x}, \mathbf{u}) = \exp(-\frac{\gamma}{2}||\mathbf{x}-\mathbf{u}||^2)$ located at each cluster center is assumed. All the clusters are assigned different weights that correspond to the fraction of the Gaussian function based on the distance from each cluster center, with $w_i = K(\mathbf{x}_i, \mathbf{u}_k)/\sum_{j=1}^{K} K(\mathbf{x}_i, \mathbf{u}_j)$, which is shown in Fig. 5.9(b).

One can also apply Fisher vector techniques. In this method, a feature space is assumed to follow a GMM. Based on the principle of a Fisher vector Perronnin *et al.* [2010], the gradient of the log-likelihood function with respect to the governing parameters is considered as the Fisher encoding of a local descriptor. With the assumption that the covariance matrix of each Gaussian mixture component is diagonal, the Fisher encoding of a set of local descriptors $X = (x_1, x_2, ..., x_N)$ extracted from an image is given by the concatenation of the following components $(\mathbf{u}_1, \mathbf{v}_1, ..., \mathbf{u}_K, \mathbf{v}_K)$.

$$\mathbf{u}_k = \frac{1}{N\sqrt{(\pi_k)}} \sum_{i=1}^{N} q_{ki} \mathbf{\Sigma}_k^{-\frac{1}{2}} (\mathbf{x}_i - \mu_k), \quad \mathbf{v}_k = \frac{1}{N\sqrt{2\pi_k}} \sum_{i=1}^{N} q_{ki} ((\mathbf{x}_i - \mu_i) \mathbf{\Sigma}_k^{-1} (\mathbf{x}_i - \mu_i) - 1) \quad (5.8)$$

Recently, sparse feature coding has been developed as an alternative for codebook learning, which is actually a generalization of k-means clustering. Given a set of local descriptors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M] \in \mathbb{R}^{D \times M}$, the k-means clustering can be reformulated as

$$\begin{aligned} \min_{\mathbf{D},\mathbf{C}} \quad & \sum_{m=1}^{M} ||\mathbf{x}_m - \mathbf{D}\mathbf{c}_m||^2 \\ \text{subject to} \quad & \|\mathbf{c}_m\|_0 = 1, \|\mathbf{c}_m\|_1 = 1, \mathbf{c}_m \succeq 0, \forall m \end{aligned} \quad (5.9)$$

where $\mathbf{D} = (\mathbf{d}_1, ..., \mathbf{d}_K) \in \mathbb{R}^{D \times K}$ is the codebook, $\mathbf{c}_m \in \mathbb{R}^K$ are the reconstruct coefficients of $\mathbf{x}_m$, and the constraints mean that there is only one non-zero element and it is 1. The index of this non-zero element is the cluster index to which it belongs. In the learning phase, both the codebook $\mathbf{D}$ and the reconstruction coefficients $\mathbf{C}$ are learned iteratively. In the coding phase, each local descriptor is assigned to its nearest cluster. It is observed that the $L0$ norm constraint $\|\mathbf{c}_m\|_0 = 1$ is too restrictive and yields only a coarse reconstruction. Yang *et al.* [2009] proposed to relax this constraint to a $L1$ norm, which gives more than one non-zero element. After relaxation, one obtains a problem of sparse coding, formulated as

$$\begin{aligned} \min_{\mathbf{D},\mathbf{C}} \quad & \sum_{m=1}^{M} ||\mathbf{x}_m - \mathbf{D}\mathbf{c}_m||^2 + \lambda ||\mathbf{c}_m||^2 \\ \text{subject to} \quad & \|\mathbf{c}_k\|_2 \leqslant 1 \quad \forall m = 1, 2, ..., k \end{aligned} \tag{5.10}$$

This optimization problem turns out to be convex with respect to one of the two parameter sets $\mathbf{D}$ and $\mathbf{C}$ when the other one is kept fixed, but the problem is not convex for both. Therefore, it can be optimized in an alternative manner. After learning the codebook, a local descriptor is encoded by the corresponding reconstruction coefficients. The word frequency histogram is replaced by a max pooling step within each region in the framework of SPM Lazebnik *et al.* [2006].

Based on the observation that non-zero coefficients are often assigned to the nearby elements in the codebook, both Yu *et al.* [2009] and Wang *et al.* [2010] proposed a local version of the sparse coding (LLC) method. In contrast to a kernel codebook, it is assumed that a local descriptor for coding can be reconstructed using the k nearest clusters. The reconstruction coefficients $w_i, i = 1, 2, ..., k$ can be computed by solving a least squares problem. The weights for the remaining clusters are set to zero. Therefore, the sparsity is replaced with locality in Eq. (5.10), which gives the following locality constrained feature coding.

$$\begin{aligned} \min_{\mathbf{D},\mathbf{C}} \quad & \sum_{m=1}^{M} ||\mathbf{x}_m - \mathbf{D}\mathbf{c}_m||^2 + \lambda ||\mathbf{s}_m \odot \mathbf{c}_m||_1 \\ \text{subject to} \quad & \mathbf{1}^T \mathbf{c}_m = 1 \quad \forall m = 1, 2, ..., k \end{aligned} \tag{5.11}$$

where $\odot$ represents the element-wise product and $\mathbf{s}_m$ is a vector of distances between the local descriptor $\mathbf{x}_m$ and all the elements in the dictionary, given as $\mathbf{s}_m = \exp(\frac{\text{dist}(\mathbf{x}_m, \mathbf{D})}{\sigma})$, $\text{dist}(\mathbf{x}_m, \mathbf{D}) = [\text{dist}(\mathbf{x}_m, \mathbf{d}_1), ..., \text{dist}(\mathbf{x}_m, \mathbf{d}_K)]$. In contrast to sparse coding, LLC has some attractive properties, like good reconstruction, local smooth sparsity, etc. One of the most important advantages is that there exists an analytical solution given by

$$\hat{\mathbf{c}}_m = \Big((\mathbf{D} - \mathbf{1}\mathbf{x}_i^T)(\mathbf{D} - \mathbf{1}\mathbf{x}_i^T)^T + \lambda \text{diag}(\mathbf{d})\Big), \quad \mathbf{c}_m = \hat{\mathbf{c}}_m / \text{sum}(\hat{\mathbf{c}}_m) \tag{5.12}$$

Thus, the LLC feature encoding can be carried out quite fast. In practice, an approximate version can be applied, which includes dictionary learning by k-means and the use of only the nearest neighbors for computing the coefficients by solving a least squares problem. All these recent feature encoding methods are evaluated by Chatfield *et al.* [2011]. Unfortunately, although the recent feature encoding methods are reportedly better than vector quantization, their improvement is very slight. Considering the computational cost and the slight improvement in accuracy, it seems better to stick to vector quantization, which works pretty well.

As the label information is not considered in codebook learning, recently some new methods for supervised dictionary learning have been proposed (Fernando *et al.* [2012]; Lazebnik & Raginsky [2009]; Mairal *et al.* [2008]; Perronnin [2008]). It seems that they might be able to provide better codebooks, but we do not consider this line of research, which might be a future direction to improve the quality of a codebook.

5.4 Three Contributions to BoW Features for SAR Image Classification

Previously, we reviewed both the local feature extraction and feature encoding methods for image classification. In this section, we present three new contributions to the BoW method for SAR image classification. The first one is using all the pixel values in a very compact neighborhood of 3×3 pixels as low level features without any additional feature extraction. We can demonstrate that vectorized local patches perform surprisingly better than other state-of-the-art features. The second contribution is that we propose to use a random dictionary instead of unsupervised clustering for dictionary learning. The elements in a random dictionary are randomly selected from the available local descriptors in a feature space. The third contribution is a new feature coding method called incremental coding, which learns a new feature representation based on the local reconstruction of other images from the same class. We demonstrated that incremental feature coding performs significantly better than other state-of-the-art feature coding methods.

5.4.1 Vectorized Patches

All the local features we reviewed above lose some information compared with the original pixel values. Since the local features represent quantities computed from the local neighborhood and there is information loss in feature extraction, it appears promising to directly use the pixel values as low level features for codebook learning. Actually, this has already been confirmed by the first experiment in section 5.2.5. In that evaluation, we used only two local statistics i.e., the mean and variance of a local patch, as low level features to learn the BoW features. Surprisingly, the method performs 5.0% better than Gabor texture features. If vectorized patches are used as low level feature vectors, a major concern could be local rotation invariance, because it is not satisfied in this case and the feature space of a class consisting of the local patches sampled from all the images might be different. However, as long as there is a sufficient number of well-separated local descriptors in the feature space, local rotation invariance would not have much influence on the classification accuracy. Therefore, the patch size should be as small as possible such that we can obtain a sufficient number of local patches. This is a bit contradictory to the motivation of Liu *et al.* [2011b, 2012], where it is claimed that the size of the sampled patches must be large enough to encompass the dominant texture variations and the importance of local rotation invariance is highlighted. In addition, in terms of simplicity and complexity, a compact neighborhood is more preferable in practice as the dimension of the feature vector increases quadratically with respect to the patch size. We demonstrate later that the sorting operation on the pixel values in a local neighborhood proposed by Liu *et al.* [2011b, 2012] performs only slightly better than vectorized patches, and that any algebraic operation on the pixel values would decrease the accuracy, like SRP Angular-diff and SRP Radial-diff as shown in Fig.5.8. Therefore, we propose to use vectorized patches as low level features without any additional feature extraction operation.

5.4.2 Random Dictionary

In the BoW method, a dictionary is usually learned by various unsupervised clustering algorithms. Dictionary learning is always taken for granted. However, in the BoW method, this step is the most time consuming. In the case of large datasets, it is prohibitively time consuming to learn a dictionary. The goal of dictionary learning is to find a universal reference for feature coding. This universal reference does not necessarily coincide with the actual cluster centers. We show that a random dictionary, collected by a random selection of some local descriptors in the feature space, is similar to one that is carefully learned by an unsupervised clustering method.

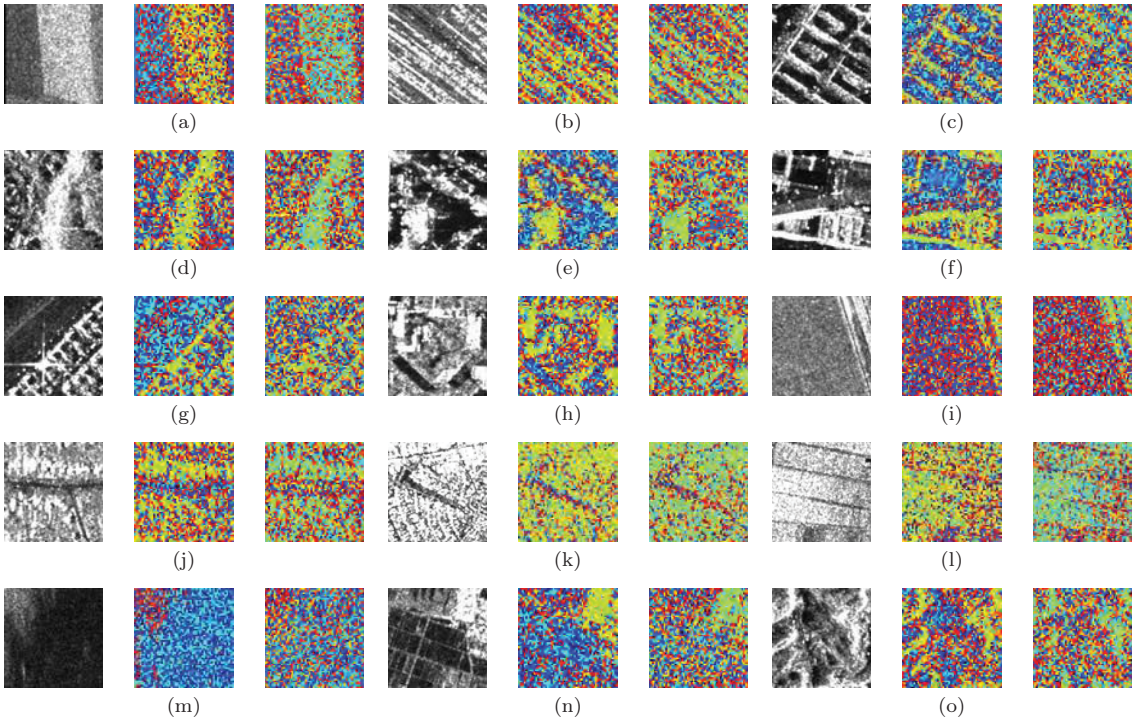


Figure 5.10: Visual comparison of vector quantization using a random dictionary and a dictionary learned by k-means clustering on our SAR dataset (see Fig. 5.13). One example is given for each of the 15 classes. The first gray scale image is an example from each class. The second and third color images in each group are the dictionary entries using a k-means dictionary and a random dictionary. Each color represents an entry of the dictionary. Both dictionaries have the same size of 200 entries.

Typical examples are given in Fig. 5.10. Here, we use the pixel values of a 3×3 vectorized patch as a local feature vector; the patches are sampled regularly from the given images. Then we compare the results of vector quantization using k-means with the results of a random dictionary. From the results of vector quantization, we see that a random dictionary can achieve similar performance as k-means. For the purpose of quantitative analysis, the vector quantization errors using both k-means and a random dictionary with the same size of 200 entries are shown in Fig. 5.11. Although the quantization error using a random dictionary is larger than for k-means, the computational cost is significantly reduced without incurring a loss in classification accuracy (see Section 5.5.6). Thus, the final feature vectors of an image are similar. This point is very important, because time consuming clustering is avoided. Thus, it makes BoW applicable and

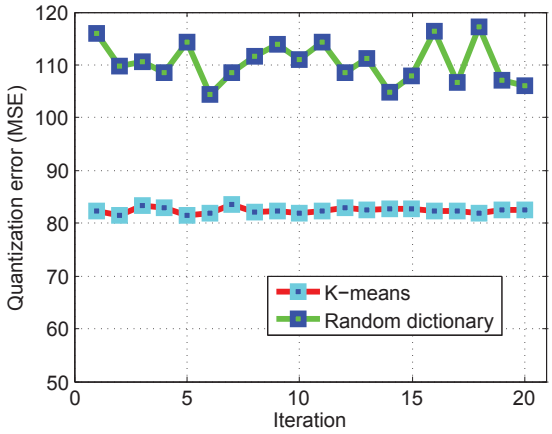


Figure 5.11: Vector quantization errors of a k-means clustering and of a random dictionary after 20 test runs of k-means clustering. The average distance of all feature vectors to their nearest neighbors is computed as a measure of the quantization error.

scalable for large databases. Similar observations have been presented by Coates & Ng [2011]. Another advantage of this method is that we do not have to load all the features into the memory. Only the random dictionary needs to be loaded into the memory. Thus, the memory requirements are significantly reduced. This is very important for large datasets, because in many cases they would probably will not fit into the memory.

5.4.3 Incremental Feature Encoding

All the feature encoding methods being reviewed in section 5.3.3 encode their local descriptors independently, which means that the images encoded before have been forgotten. In other words, the methods do not consider the relation between the local descriptors from the same class. This is obviously contradictory to human learning. For instance, when you see a dog for the first time, you are told that this is a dog and you can encode the dog using simple features. After seeing dogs for many times, obviously you should encode a dog better than for the first time based on the previous knowledge learned about dogs. Analogously, a feature encoding algorithm should encode the local features in an incremental manner. That means the algorithm should encode a new image based on the previously encoded images. In view of this motivation, we propose an incremental feature encoding based on the LLC algorithm, as shown in Fig. 5.12. The main difference from previous encoding methods is that the encoding of a new image depends on the previously encoded images, and the encoding is performed in an incremental manner. The algorithm starts from a database, and a dictionary is learned using k-means clustering. The feature encoding is done class by class. The encoding starts from the first image in one class, and the nearest neighbors are searched. After that, the local reconstruction coefficients $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_N]$ (N is the number of local features extracted from one image) are obtained by solving a least squares problem. Based on the reconstruction coefficients of the previous image, the encoding of the next image is performed by replacing only a few coefficients in each column of $\mathbf{C}$ because only the nearest neighbors are involved in the computation of the coefficients. Therefore, the reconstruction coefficients are learned in an incremental manner. For each image, the final feature vector of an image is extracted by max pooling $\max([\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_N])$, which selects the maximum value in each row. This incremental feature encoding has shown a significantly better performance than other feature encoding methods.

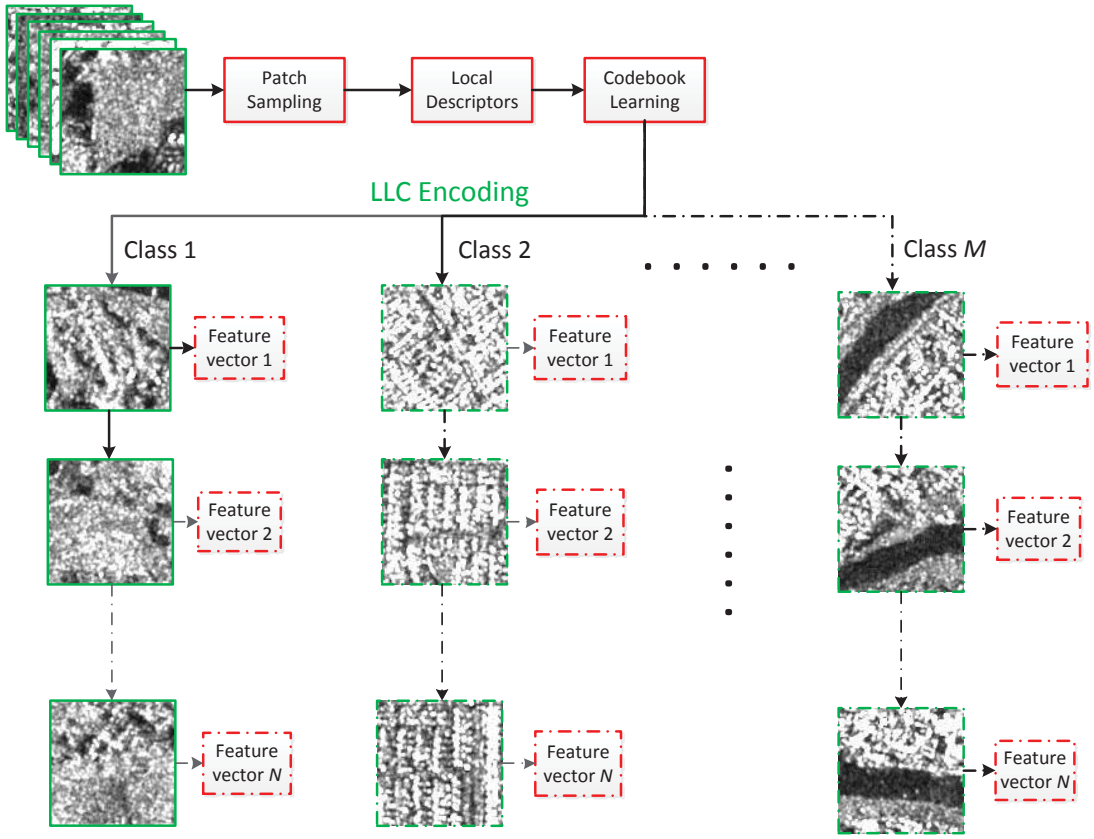


Figure 5.12: Incremental feature coding.

5.5 Evaluation Results and Comparisons

In this section, a rigorous evaluation of the following open problems in the BoW model for SAR image classification is carried out.

1. Which is the best local patch size and the best patch sampling strategy?
2. Which local features should be extracted?
3. What is the strategy of dictionary learning within the BoW framework? A universal dictionary or multiple class-specific dictionaries?
4. Involved sparse coding or simple vector quantization?
5. Nearest neighbor assignment or multiple assignments?

All these problems have to be tackled with care. If we optimize all these components, a simple unsupervised feature learning algorithm could be able to achieve state-of-the-art accuracy. This has been observed by Coates & Ng [2012].

We collected 15 classes from a total of 3434 TerraSAR-X images with a size of 160×160 pixels as shown in Fig. 5.13. The images are radiometrically enhanced high resolution Stripmap TerraSAR-X products. Their pixel spacing is about 1.9 m. This dataset is collected using an active learning system Cui *et al.* [2013b]. Among the 15 classes, there are 7 classes chosen

from urban areas, which is sufficient to evaluate methods for urban area classification. There are three classes related to agricultural fields. One class refers to flooded fields. The last class contains mountainous areas. The classifier used for evaluation in the following is a C-SVM with a polynomial kernel function. In each round, we randomly selected 30 samples from each class as training data and report the average accuracy of 20 rounds.

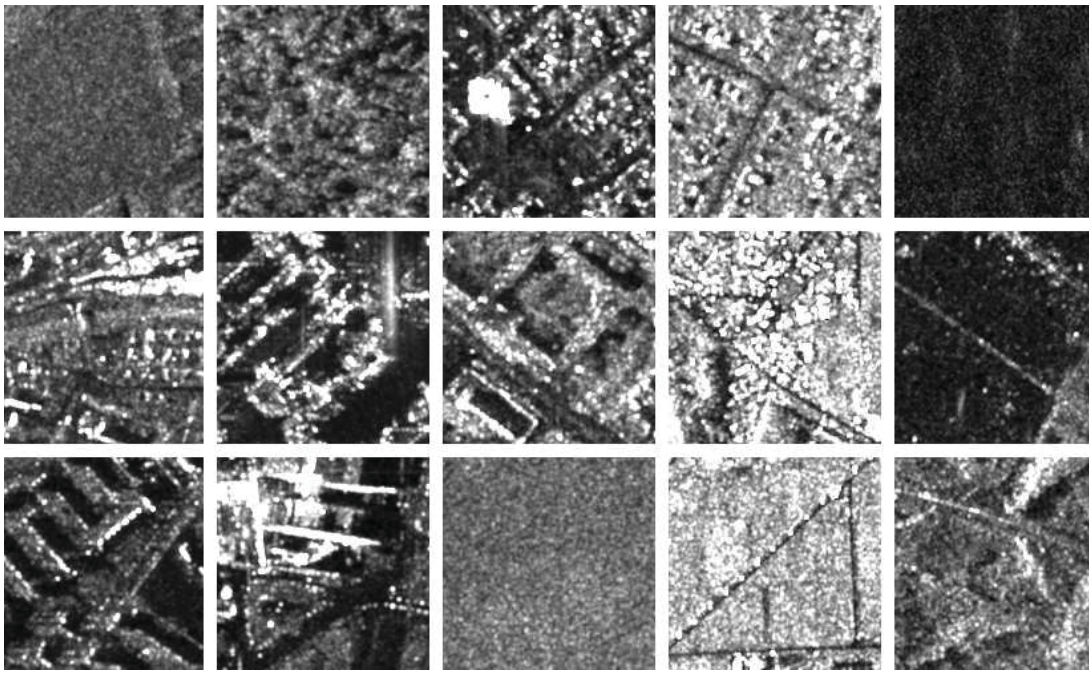
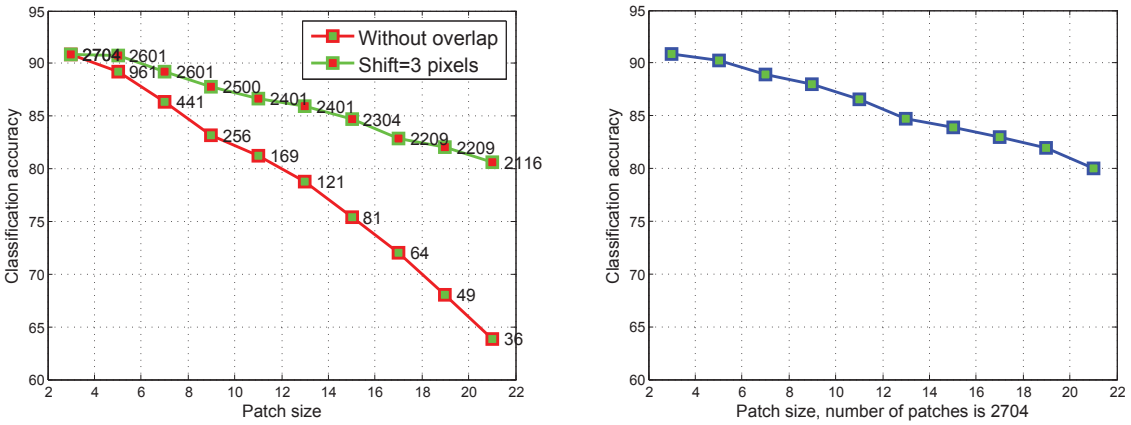


Figure 5.13: All 15 classes of 3434 TerraSAR-X images to be used for evaluation with a size of 160×160 pixels. Among them 7 classes are collected from urban areas. The number of images in each class is respectively 319, 195, 158, 118, 324, 296, 211, 151, 430, 123, 172, 204, 134, 254, and 345.

5.5.1 Local Patch Size

In the following, we used different window sizes varying from 3×3 to 21×21 pixels for patch sampling. The resulting local features are the vectorized pixel values in the local neighborhood and the dictionary size is kept fixed at 200 entries. Vector quantization is employed to learn the dictionary since it is the most widely used method. To this end, each image is split into patches of a predefined size. Three evaluations are performed. In the first evaluation, we do not allow any overlap between patches, thus the number of patches decreases as the patch size increases. The classification accuracy versus patch size is shown as the red curve in Fig. 5.14(a). The number besides each marked point on the curve is the number of patches having been used. Obviously, the accuracy decreases as the patch size increases. A patch size of 3×3 pixels is the best one. Since the number of patches sampled in one image decreases as the patch size increases, this might be a reason for the decreasing accuracy, rather than the patch size itself. Therefore, in the second evaluation we allow some overlapping between patches in order to increase the number of patches that can be sampled from an image. The resulting classification accuracy versus patch size is plotted as the green curve in Fig. 5.14(a). Similar to the first evaluation,

the accuracy decreases as the patch size increases, which is consistent with the observation of the first evaluation. Although we allow overlaps between nearby patches, the number of patches still continues to decrease as the patch size increases, which can be seen from the numbers being depicted besides the marked points. In the last evaluation, we keep the number of patches constant and we randomly sample a fixed number of 2704 patches from each image, which is the maximum number of patches with a size 3×3 pixels in the case of no overlap. The corresponding accuracies are presented in Fig. 5.14(b). Apparently, the accuracy still decreases as the patch size increases. In addition, the resulting blue curve is very similar to the green curve in Fig. 5.14(a). From this observation, we can understand that even if we increase the number of patches for large patch sizes, there is still not much improvement in accuracy. Therefore, we can safely draw the conclusion that a compact neighborhood size 3×3 pixels is better than large patch sizes. This contradicts the claim by Liu *et al.* [2011b] that the patch size must be large enough to encompass the dominant texture variations. With smaller patch sizes, the image content variation can still be captured by the word histogram in the BoW framework, because the word histogram counts the number of clusters that occur in the image. Thus, a smaller patch size is still able to capture a large variation in the image content by the word histogram. The reason that the accuracy of a large patch size is worse can be explained by the fact that the feature space in the case of a large patch size cannot be well separated. In addition, for large patch sizes, there is a large overlap between adjacent patches. Although the number of patches is sufficient due to the large overlap, the patches might not be representative enough to form clusters in the feature space. Consequently, the final word histogram is less discriminative.



(a) (b)

Figure 5.14: Patch size evaluation: (a) Classification accuracy versus patch size using regular patch sampling. The red curve represents the case without overlap, and the image is cut regularly with a fixed patch size. The number besides each marked point on the curve stands for the number of patches. The green curve refers to the case with a 3 pixel horizontal and vertical shift. (b) Classification accuracy versus patch size using random sampling to keep a fixed number of 2704 patches.

5.5.2 Sampling Strategy

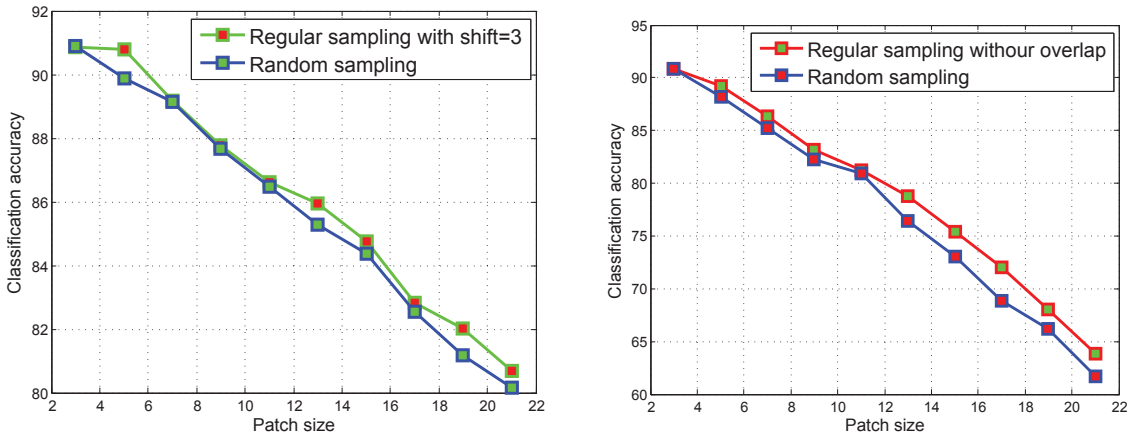
Although the BoW model representing an image as a collection of independent local patches has shown very powerful ability for image classification, there is still one important open issue to be

answered, which is what patches to sample from a given image. Due to the fact that representative patches may occur at any position and scale, dense sampling can capture most information. But, in practice, it is not realistic and applicable because of limited memory capacity and the computational effort. We think that computational effort can be reduced while preserving the accuracy by densely sampling all patches with a very small patch size. To evaluate this argument, we compare regular dense sampling with and without overlap and random sampling of patches with different sizes while keeping fixed the number of patches. The row and column positions of the patches are determined by random samples drawn from a uniform distribution. Regular dense sampling with overlap corresponds to the green curve in Fig. 5.14(a). The number of patches is fixed for a given patch size, but the sampling strategy is replaced with random sampling. The attained classification accuracy versus patch size is plotted in Fig. 5.15(a). It can be clearly seen that although regular dense sampling is slightly better than random sampling, there is no big difference as long as the number of patches remains the same and the entire image can be fully covered. In a second evaluation, we compare random sampling with regular dense sampling without overlap while keeping the number of patches fixed. The resulting accuracy versus patch size is shown in Fig. 5.15(b). Regular sampling is also slightly better than random sampling because random sampling cannot fully cover the entire image. To obtain a clear answer to the influence of the sampling strategy, we verify the impact of increasing the number of patches that are randomly sampled from a given image with small and large patch sizes. The effect of increasing the number of patches in the case of dense sampling is shown in Fig. 5.16 for patch size 3 (a) and 11 (b). For large patch sizes, the accuracy increases in parallel with the number of patches. However, the accuracy becomes stable beyond 4000 patches, which means there is no additional gain in accuracy by continuously increasing the number of patches because a large number of patches are duplicated in the case of random sampling, which can prohibit a good clustering. In addition, large patch sizes will increase the computational burden. For smaller patches, the accuracy has a similar behavior as shown in Fig. 5.16(b). However, it should be noted that increasing the number of large patches with random sampling cannot reach the accuracy of regular dense sampling with a small patch size.

Therefore, we conclude that two conditions have to be satisfied for patch sampling: The first one is that the entire image needs to be covered by the patches. The second one is that the number of patches has to be sufficiently high such that they form a well-separated feature space. Regular dense sampling with a small patch size has a slightly better accuracy than random sampling with the same number of patches. For large patch sizes, increasing the number of patches with random sampling can improve the accuracy, but this approach is inferior to regular sampling with small patches, which confirms the conclusion of the previous evaluation.

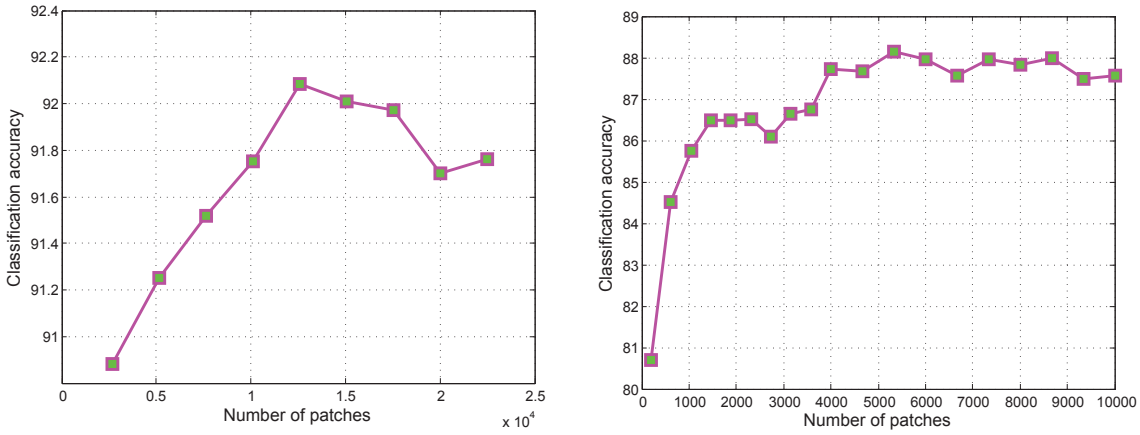
5.5.3 Dictionary Size

A dictionary obtained through k-means has a potential influence on the attainable accuracy. Different dictionary sizes are tested to evaluate this impact. In the case of 3×3 patches, the resulting accuracy versus dictionary size is shown in Fig. 5.17(a). It can be clearly seen that the accuracy depends on the dictionary size because a small dictionary cannot capture the distribution of the feature space and is underfitting the model. However, the accuracy reaches its peak when the dictionary size is around 250 entries. Beyond that value, the accuracy decreases again because the feature space is overfit with a large dictionary. In addition, the computation time versus dictionary size is plotted in Fig, 5.17(b). The time increases linearly in parallel with the dictionary size. Therefore, for a practical application, it would be better to choose an appropriate dictionary size in terms of both accuracy and computational effort. In this case, it



(a) (b)

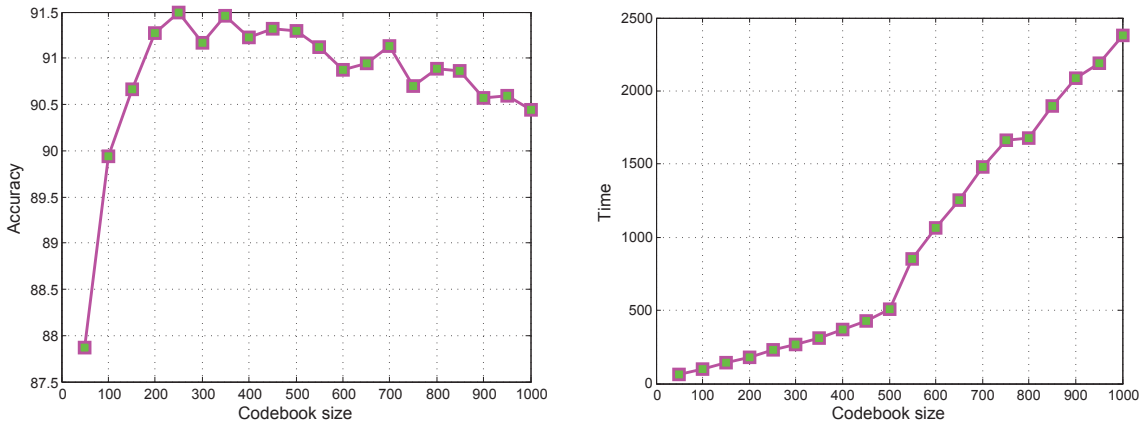
Figure 5.15: Evaluation of regular dense sampling and random sampling with the same number of patches: (a) Comparison of regular dense sampling with random sampling and the same number of patches as in the case with overlap. In this case, the entire image can be fully covered. (b) Comparison of regular dense sampling with random sampling and the same number of patches as in the case without overlap. In this case, the entire image might not be fully covered by random sampling.



(a) (b)

Figure 5.16: Impact of the number of patches sampled from an image for large and small patch sizes: (a) Impact of the number of patches sampled from an image with a size of 3×3 pixels. (b) Impact of the number of patches sampled from an image with a patch size of 11×11 pixels. The first abscissa point corresponds to the maximum number of patches that we can obtain by regular sampling from an image. When we increase the number of 3×3 patches by random sampling, the accuracy reaches a peak at around 4 times the maximum number of patches with regular dense sampling, which is much less than the maximum number of patches that can be sampled from an image. This implies that increasing the number of patches with random sampling can increase the accuracy slightly; however, it is not necessary to use all patches of an image.

seems a size of 250 entries is a good choice.



(a) (b)

Figure 5.17: Impact of dictionary size on the classification accuracy and time computation: (a) the effect of dictionary size; (b) the computational time.

5.5.4 Universal Dictionaries vs. Class-Specific Dictionaries

One important issue in learning a dictionary is to construct either a universal dictionary for all classes or a concatenation of class-specific dictionaries. For the method of texton generation reviewed in section 3.3, the texton dictionary is first generated for a single class and then combined as a universal dictionary for all classes. In this subsection, we evaluate this problem and give a clear answer. Therefore, we keep the dictionary size in these two cases fixed and plot the accuracy versus dictionary size in Fig. 5.18. It can be seen clearly that the universal dictionary is always better than a concatenation of class-specific dictionaries. The reason is that different classes might have some clusters in common. The universal dictionary can capture the feature distribution without considering the distribution of each class. In contrast, class specific dictionaries might have some common clusters but they are considered separately; thus their feature space is prone to underfitting. However, the computational effort to obtain a universal dictionary is much more time consuming than generating a concatenation of class-specific dictionaries as a large volume of feature vectors is involved in the clustering, in addition to the curse of dimensionality.

5.5.5 Local Feature Extraction

From the previous sections, we conclude that the pixel values of a very compact patch 3×3 patch can achieve very promising accuracy with a medium dictionary size. Therefore, we compare this as a baseline with seven other feature extraction methods presented in section 5.3.1.2, i.e., RIFT, SPIN, and five sorted features. In the following three evaluations, we vary a single parameter while keeping the remaining parameters fixed .

In the first evaluation, we fix the dictionary size to 200 and evaluate the classification accuracy using different patch sizes, ranging from 3×3 to 21×21 pixels with a shift of 3 pixels in horizontal and vertical directions. The resulting accuracy with different patch sizes is shown in Fig. 5.19, where the vectorized patch feature is taken as a baseline for comparison. It is interesting to see that sorting the pixel values in the local neighborhood gives promising results, although the

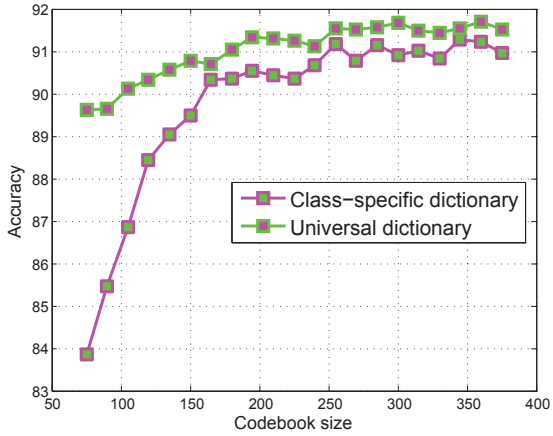


Figure 5.18: Evaluation of universal and class-specific dictionaries.

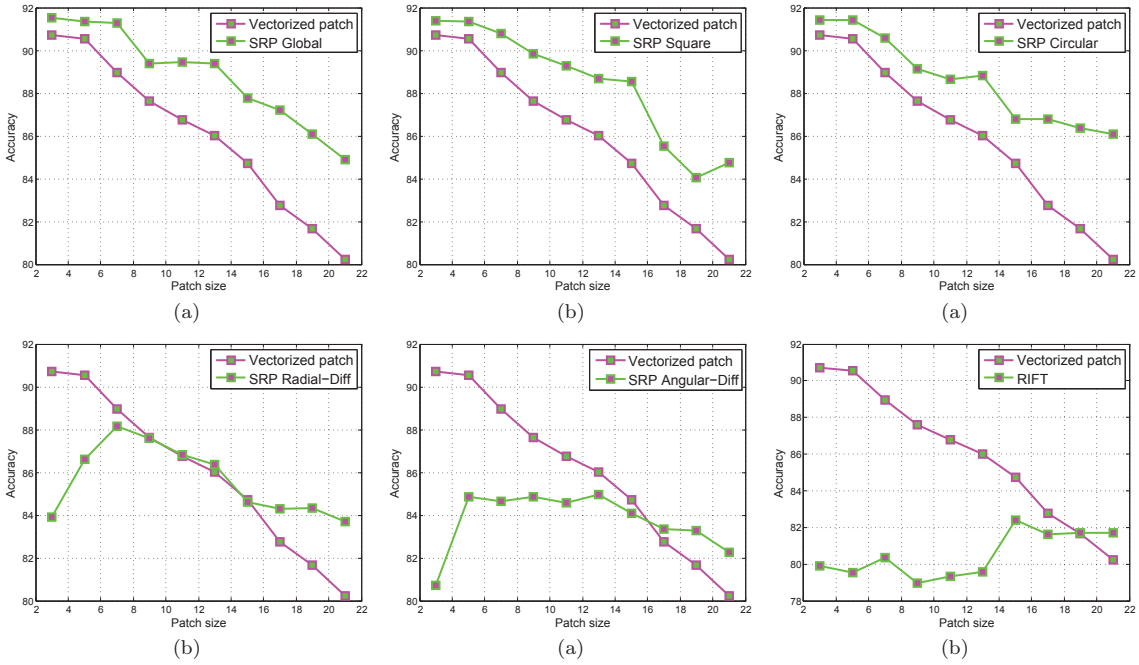


Figure 5.19: Evaluation of local features for SAR image classification using different patch sizes and a fixed dictionary size.

improvement over the vectorized patch feature is slight for a patch size of 3×3 pixels. The first three feature extraction methods, namely SRP Global, SRP Square, and SRP Circular show a similar behavior when the patch size increases: the accuracies of these three cases decreases when the patch size increases; this confirms our conclusion drawn previously in section 5.14. In contrast, the SRP Radial-diff method shows increasing accuracy when the patch size increases but it remains inferior to vectorized patches. SRP Radial-diff reaches an accuracy peak for a patch size of 7×7 pixels. Beyond that size, the accuracy decreases again when the patch size increases and it remains inferior to vectorized patches. SRP Angular-diff shows a sharp improvement in accuracy when changing from a patch size of 3×3 pixels to 5×5 pixels. For

patch sizes between 7×7 and 13×13 pixels, the method has a performance similar to vectorized patches. For smaller patch sizes, vectorized patches perform much better than SRP Radial-diff and SRP Angular-diff. Unfortunately, RIFT performs quite poorly compared to other feature extractors. From this evaluation, we can conclude that using all the pixel values in a patch gives a good accuracy. However, any algebraic operations on the patch can reduce the accuracy.

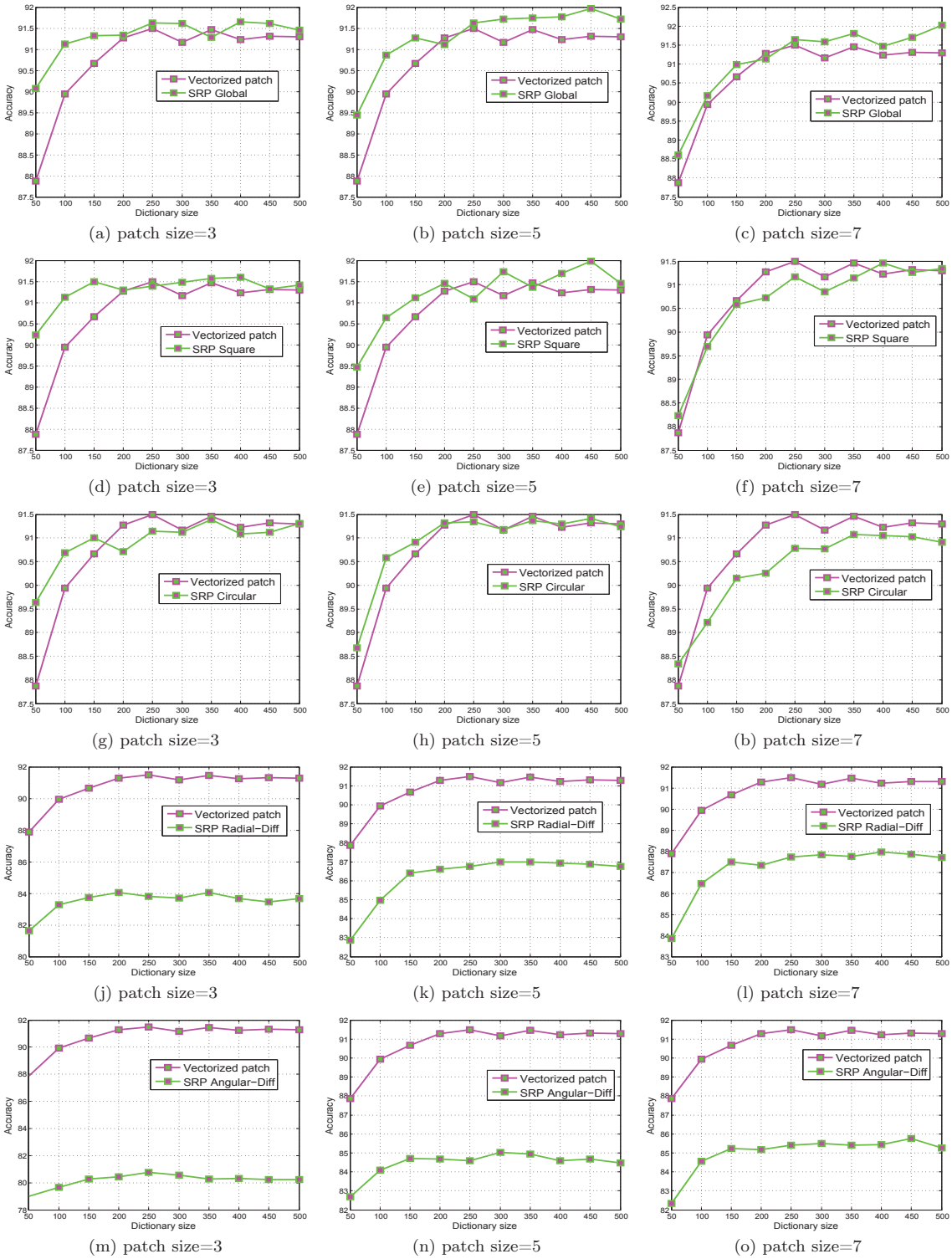
In our second evaluation, we fix the patch size to 3×3, 5×5 and 7×7 pixels and vary the dictionary size from 50 to 500 entries, in order to get a better understanding of the feature extractors. In this comparison, our baseline is defined by the vectorized pixel values of a 3×3 patch. The classification accuracy versus dictionary size is shown in Fig. 5.20. Each column shows the comparison of a selected method with our baseline approach when using different patch sizes. Generally, the accuracy is higher for large dictionaries but becomes stable with a sufficiently large size, which is consistent with the conclusion drawn previously in section 5.17. It can be seen clearly from the first three rows that the first three methods do not differ much from the vectorized patches for a sufficiently large dictionary. On the contrary, the vectorized patch approach performs much better than SRP Radial-diff and SRP Angular-diff for all dictionary sizes. Any algebraic operations on the pixel values in the neighborhood results in some information loss. In addition, the difference between two pixel values depends on the global image brightness.

In a third evaluation, we vary the number of training samples while keeping the dictionary size and the patch size fixed. The accuracy of all five methods with a varying number of training samples is shown in Fig. 5.21. Obviously, SRP Angular-diff and Radial-diff are inferior to the remaining four methods. The common characteristic of all these four methods is that they use all pixel values of a local neighborhood. From this, we conclude that any operations that cause some information loss should be avoided. Another observation is that the accuracy differences are negligible for very small patch sizes. When we increase the patch size, the advantage of a sorting operation becomes obvious. However, the accuracy decreases when the patch size increases. Therefore, aiming at the best accuracy, we should use all the pixel values within a very small patch as low level features. In addition, the advantage of SRP Global becomes obvious when the patch size increases compared with SRP Square, SRP Circular, and vectorized patches.

We conclude that vectorized patches, without extracting any features, perform quite well in the framework of a BoW model for SAR image classification. Sorting the pixel values in the compact patch performs slightly better but does not give much improvement. However, for small patches, SRP Angular-diff and SRP Radial-diff are always inferior to vectorized patches. On the other hand, in terms of the computational effort, SRP circular, SRP Angular-diff and SRP Radial-diff are more time consuming than vectorized patches because they require interpolations at non-integer positions. From a practical point of view, vectorized patches are a good choice for large scale applications.

5.5.6 Learned Dictionary or Random Dictionary

In this experiment, we compare random dictionary learning and k-means dictionary learning in terms of classification accuracy. Three evaluations are performed. In the first evaluation, we use the vectorized patch of a 3×3 pixel window as a low level feature vector. The elements in the random dictionary are randomly selected from all the local feature vectors. The classification accuracy versus dictionary size is shown in Fig. 5.22(a). We can clearly see that there is not much difference between a random dictionary and the one learned using k-means. In the case of large dictionaries, a random dictionary is even better. This is very important for practical applications as dictionary learning using k-means is usually quite time consuming and may



(a) patch size=3 (b) patch size=5 (c) patch size=7

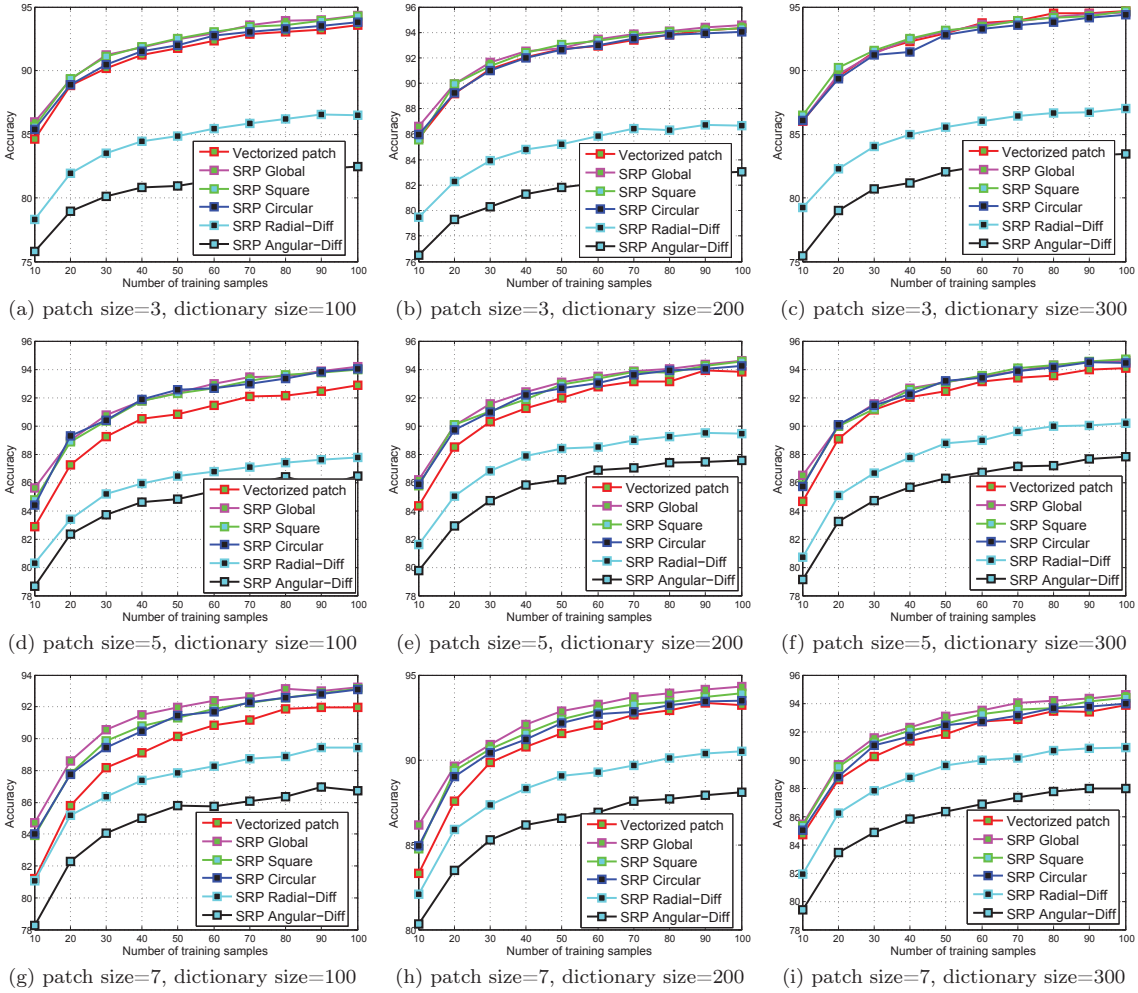
(d) patch size=3 (e) patch size=5 (f) patch size=7

(g) patch size=3 (h) patch size=5 (b) patch size=7

(j) patch size=3 (k) patch size=5 (l) patch size=7

(m) patch size=3 (n) patch size=5 (o) patch size=7

Figure 5.20: Evaluation of local feature extractors with different dictionary sizes.

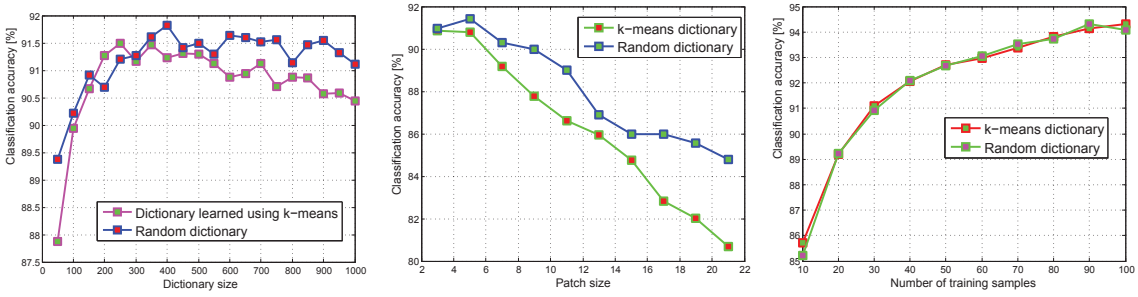


(a) patch size=3, dictionary size=100 (b) patch size=3, dictionary size=200 (c) patch size=3, dictionary size=300

(d) patch size=5, dictionary size=100 (e) patch size=5, dictionary size=200 (f) patch size=5, dictionary size=300

(g) patch size=7, dictionary size=100 (h) patch size=7, dictionary size=200 (i) patch size=7, dictionary size=300

Figure 5.21: Evaluation of local feature extractors using a varying number of training samples.



(a) (b) (c)

Figure 5.22: Comparison of a random dictionary with a dictionary learned using k-means: (a) Comparison using different dictionary sizes; (b) Comparison using different patch sizes; (c) Comparison using different numbers of training samples.

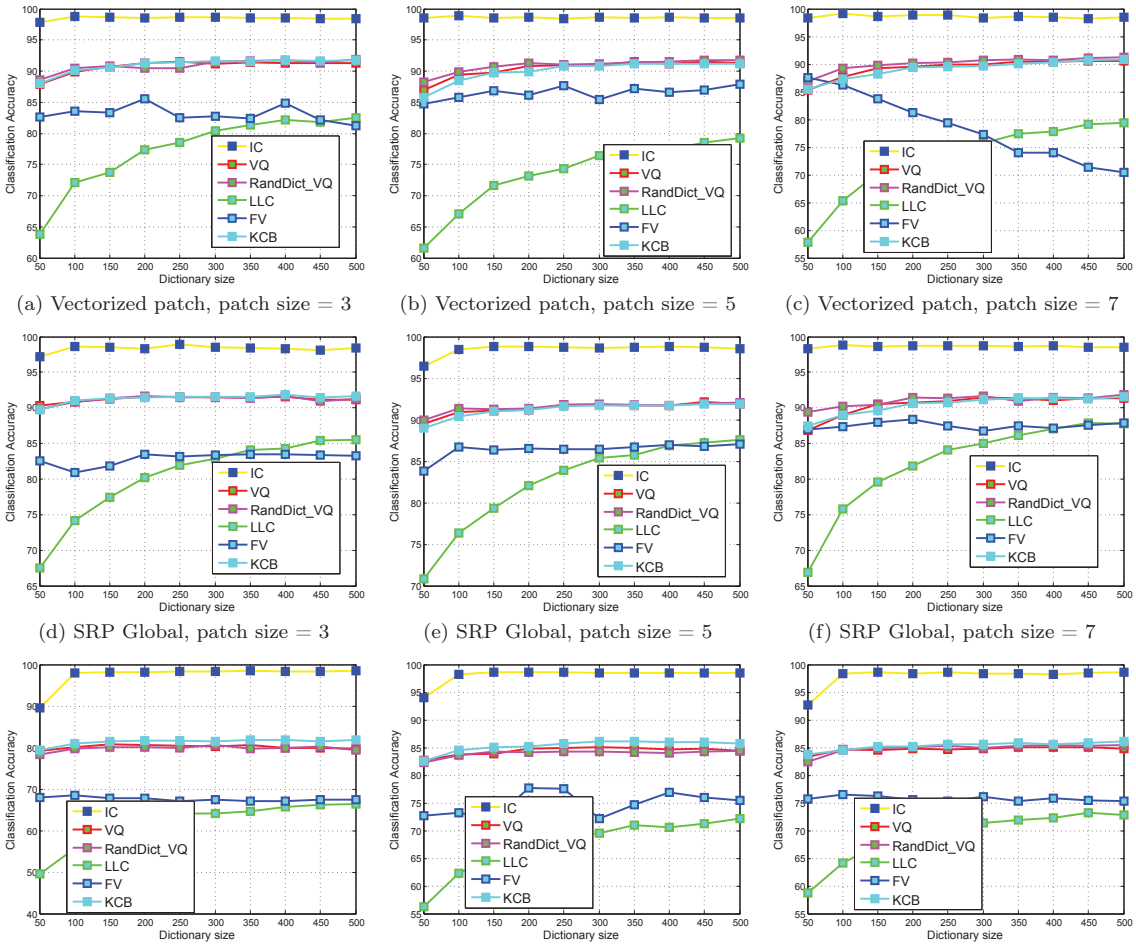
become prohibitively slow for large datasets. From this evaluation, we see that it is not necessary to spend time for learning a dictionary using unsupervised learning methods. As long as the elements in the dictionary can provide full support for the data points in the feature space, even a random dictionary can give very good accuracy.

In the second evaluation, we investigate the performance of a random dictionary versus patch size. We fix the dictionary size to 200 entries and vary the patch size from 3×3 to 21×21 pixels with a 3 pixel window shift in two directions. The classification accuracy versus patch size is shown in Fig. 5.22(b). It becomes evident that a random dictionary is superior to dictionaries learned by k-means. In the last evaluation, we change the number of training samples while keeping fixed the patch size of 3×3 pixels and the dictionary size of 200 entries. The classification accuracy versus the number of training samples is shown in Fig. 5.22(c). We can clearly observe that they are almost the same. Through these three evaluations, we conclude that a random dictionary can achieve a good performance and, in some cases, an even better accuracy than k-means.

5.5.7 Sparse Coding or Vector Quantization

In section 5.3.2 we presented several feature encoding methods for image classification, such as vector quantization (VQ) Sivic & Zisserman [2003], Fisher vector encoding (FV) Perronnin *et al.* [2010], kernel codebook technique (KCB) van Gemert *et al.* [2010], and locality constrained coding (LLC) Wang *et al.* [2010]. In this section, we compare our incremental coding (IC) with other methods and provide a comprehensive evaluation of their performances. In addition, we also plot the results of vector quantization using a random dictionary (denoted by RandDict in the following figures). In this evaluation, we used varying patch sizes of 3×3, 5×5, and 7×7 pixels and 10 dictionary sizes ranging from 50 to 500 entries. As already shown in section 5.5.5, SRP Global performs slightly better than vectorized patches and SRP Angular-diff performs worse; thus, we use only these latter three local feature extraction methods for evaluating feature encoding techniques.

All the results are shown in Fig. 5.23. The first row shows the performance comparison of all the six feature encoding methods using vectorized patches and three different patch sizes, while the comparisons in the second and third row use SRP Global and SRP Angular-diff respectively. It can be immediately seen that our incremental feature encoding method does not only provide significantly better results than the other methods, but is also stable for different dictionary sizes because the feature vectors are learned based on all previous images. This is a new perspective for feature learning, as the conventional methods always assume that the local feature vectors are independently drawn from some underlying distribution. Actually, each class should be considered as an entity in the feature space, but not as a collection of feature vectors. It is worth noting that incremental feature coding can achieve high accuracies even for a less discriminative method, like SRP Angular-diff shown in the third row. Vector quantization ranks second and is actually a very good option, although there are many methods trying to improve its performance. Although the use of a kernel codebook was proposed to overcome the drawback of vector quantization, its actual improvement is negligible. It is only slightly better than vector quantization for the less discriminative SRP Angular-Diff. Therefore, it seems that there is no gain in accuracy by assigning a local feature vector to multiple neighbors. The performances of both vector quantization and kernel codebook are quite stable with respect to the dictionary size. It is interesting to see that both LLC and FV perform worse than vector quantization. FV performs even worse, but its accuracy remains quite stable versus dictionary size. The most devastating characteristic of FV is that it is very time consuming to learn a mixture model in the



(a) Vectorized patch, patch size = 3 (b) Vectorized patch, patch size = 5 (c) Vectorized patch, patch size = 7

(d) SRP Global, patch size = 3 (e) SRP Global, patch size = 5 (f) SRP Global, patch size = 7

(g) SRP Angular-diff, patch size = 3 (h) SRP Angular-diff, patch size = 5 (i) SRP Angular-diff, patch size = 7

Figure 5.23: Evaluation of feature encoding methods for SAR image classification. The comparisons in the first row use vectorized patches as low level features while the second and the third row use SRP Global and SRP Angular-diff.

case of a large dictionary. The method with the poorest performance is LLC, whose performance improves with increasing dictionary size. From the accuracies shown in Fig. 5.23, we can see that both the local feature descriptor and the feature coding method are very important. Bad choices of these components will reduce the overall performance.

We conclude that although there are many methods trying to improve vector quantization by reducing the information loss, there is not much gain in accuracy. Our incremental feature coding method achieves significantly better results than state-of-the-art methods, even in the case of significantly less discriminative features.

5.5.8 Comparison with State-of-the-Art Methods

In this section, we compare the BoW method with state-of-the-art feature extraction methods, where 3×3 vectorized patches are used as low level features followed by vector quantization and incremental feature coding respectively, using a random dictionary and another one learned by

k-means. The four BoW methods are abbreviated as KmDict_VQ, KmDIct_IC, RandDict_VQ, and RandDict_IC in the following. The state-of-the-art feature extractors we selected for comparison are based on Gabor texture, GLCM texture, wavelet texture, STFT and QMF.

- Gabor texture features contain the statistics of the Gabor filter responses. Gabor filters are characterized by their scale and orientation. We compare our method with two sets of statistics computed using the filter responses. The first set contains the mean and variance of each sub-band as proposed by Manjunath & Ma [1996], while the second set consists of the log-means and log-variances of all sub-bands, which have been demonstrated as superior for SAR image retrieval by Singh & Datcu [2013]. The number of scales and orientations are set to 4 and 6 respectively. Thus, the dimension of the feature vector is $4 \times 6 \times 2 = 48$.

- GLCM texture features as published by Haralick *et al.* [1973] are the statistics of the co-occurrence matrix, which contains the second order statistics of pairs of pixels with a certain number of horizontal and vertical pixel shifts. To reduce the computational complexity, an additional quantization of the gray levels is usually applied when constructing the co-occurrence matrix. As suggested by Clausi [2002], setting the number of levels to a value of less than 24 can produce unreliable classification results, while an excessive number of levels (greater than 64) is unnecessary since it does not improve the classification accuracy and is computationally costly. Therefore, we set the number of quantization levels to 32. The number of orientations is set to 4, and the number of shifts runs from 1 to 4. The statistics we computed are autocorrelation, contrast, correlation, cluster prominence, cluster shade, dissimilarity, energy, entropy, homogeneity, maximum probability, sum of squares, sum average, sum variance, sum entropy, difference variance, difference entropy, information measure of correlation, inverse difference, normalized inverse difference, and normalized inverse difference moment. In total, the dimension of the feature vector is $20 \times 4 \times 4 = 320$, which is independent of the number of quantization levels.

- The texture features derived from a wavelet transformation are the statistics, i.e., mean and variance, of the filter bank responses. An image is decomposed into 3 levels using both a non-decimated 2D wavelet transformation (NDWT) as well as a dual tree complex wavelet transformation (DTCWT) proposed by Selesnick *et al.* [2005]. Similar to the Gabor texture features, two versions of features are computed. For the non-decimated 2d wavelet transform, a Daubechies filter Strang & Nguyen [1997] is applied, while for the dual tree complex wavelet transformation, near-symmetric 13,19 tap filters are used for the first level and Q-Shift 14,14 tap filters are employed for the higher levels. The dimensions of the two feature vectors are 18 and 36 respectively.

- STFT features as proposed by Popescu *et al.* [2008] are 6 parameters based on a short term Fourier transformation, which include the mean and variance, the spectral centroid and the spectral flux in horizontal and vertical direction. The spectral centroid is the centroid of a short-term Fourier transformation and is a measure of the spectral brightness.

- QMF features of Simoncelli & Adelson [1990] are nothing but the means and variances of all sub-bands in the image pyramid. The number of levels is set to 3, thus, there is only one low pass band and 3 horizontal sub-bands, 3 vertical sub-bands, and 3 diagonal sub-bands in the sub-band pyramid. Therefore, the dimension of the feature vector is $(1 + 3 + 3 + 3) \times 2 = 20$.

- Fractional Fourier transformation (frFFT) features were proposed by Singh & Datcu [2012, 2013] for SAR image classification. The log-moment and log-variance of all sub-bands are

used as a feature vector to characterize a SAR image. The only selectable frFFT parameter is the number of angles, which is assumed to be 18 in this comparison. Therefore, the corresponding feature vector dimension is $18 \times 2 = 36$.

Table 5.1: Performance comparison with state-of-the-art features.

Features	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Mean
Gabor	0.84	0.78	0.54	0.33	0.68	0.80	0.55	0.44	0.84	0.57	0.82	0.76	1.00	0.59	0.73	0.70
log-Gabor	0.90	0.83	0.72	0.54	0.78	0.87	0.84	0.61	0.88	0.91	1.00	0.99	1.00	0.92	0.88	0.85
GLCM	0.76	0.71	0.50	0.29	0.61	0.66	0.70	0.47	0.74	0.84	0.89	0.71	0.90	0.56	0.65	0.67
NDWT	0.64	0.68	0.57	0.24	0.57	0.75	0.46	0.30	0.84	0.56	0.79	0.86	0.99	0.70	0.68	0.66
log-NDWT	0.59	0.82	0.46	0.32	0.64	0.85	0.60	0.52	0.92	0.90	0.99	0.98	1.00	0.86	0.65	0.74
DTCWT	0.73	0.77	0.57	0.32	0.70	0.79	0.59	0.36	0.85	0.59	0.84	0.81	1.00	0.68	0.72	0.71
log-DTCWT	0.85	0.84	0.76	0.51	0.83	0.87	0.85	0.61	0.93	0.92	1.00	0.99	1.00	0.92	0.87	0.86
STFT	0.45	0.20	0.12	0.01	0.05	0.26	0.01	0.10	0.49	0.01	0.18	0.37	1.00	0.01	0.39	0.28
QMF	0.67	0.66	0.59	0.23	0.61	0.70	0.67	0.30	0.92	0.68	0.85	0.81	0.99	0.81	0.71	0.69
frFFT	0.81	0.85	0.80	0.44	0.82	0.88	0.87	0.58	0.94	0.82	0.96	0.97	1.00	0.92	0.88	0.85
KmDict_VQ	0.96	0.93	0.85	0.63	0.89	0.95	0.95	0.88	0.97	0.97	0.99	0.96	0.99	0.91	0.94	0.91
RandDict_VQ	0.90	0.89	0.95	0.82	0.86	0.99	0.95	0.71	0.91	0.86	1.00	0.95	1.00	0.92	0.91	0.91
KmDict_IC	1.00	0.99	0.99	0.98	1.00	0.99	0.99	1.00	0.99	0.99	0.99	1.00	1.00	0.99	0.99	0.99
RandDict_IC	1.00	0.99	0.99	1.00	0.97	0.99	0.99	0.98	0.99	0.99	1.00	1.00	0.98	0.87	1.00	0.98

The accuracies of the selected feature extraction methods for all 15 classes are shown in Table. 5.1. It can be clearly seen that the four BoW methods perform significantly better than the other approaches, with an average accuracy of more than 90%. The average accuracies of all other methods are lower than 90%. The average accuracy of incremental coding is almost one hundred percent, which is much better than the other feature encoding methods. For both incremental coding and vector quantization, we can see that a random dictionary can play a same role as a dictionary learned using k-means clustering. Vector quantization ranks second, followed by log-Gabor and log-DTCWT with similar performances. frFFT performs better than the remaining methods. In addition, we can see that the log versions of Gabor, NDWT, and DTCWT perform much better than the linear versions. STFT features rank last; the reason for this could be the lower dimension of the feature vector. In this evaluation, 30 images are selected from each class and used as training samples. The remaining images are used as test data. The effects of increasing the number of training patches on the classification accuracies of all selected methods are shown in Fig. 5.24. It can also be seen that the BoW results are much better than all other methods with accuracies of less than 90%. Incremental feature coding has the best accuracy and comes close to one hundred percent.

5.5.9 Summary

In this section, we propose a simple yet efficient feature extraction method within the Bag-of-Words (BoW) framework. It has three main innovations. Firstly, It is interesting that using only the pixel values within a very small neighborhood like 3×3 pixels without any additional feature extraction gives a much better performance than many complex methods with high computational complexity. Secondly, in contrast to many unsupervised feature learning methods, a random dictionary is applied to feature space quantization. The advantage of a random dictionary is that it does not lead to a significant loss of classification accuracy yet the time-consuming process of dictionary learning is avoided. These two novel improvements over state-of-the-art

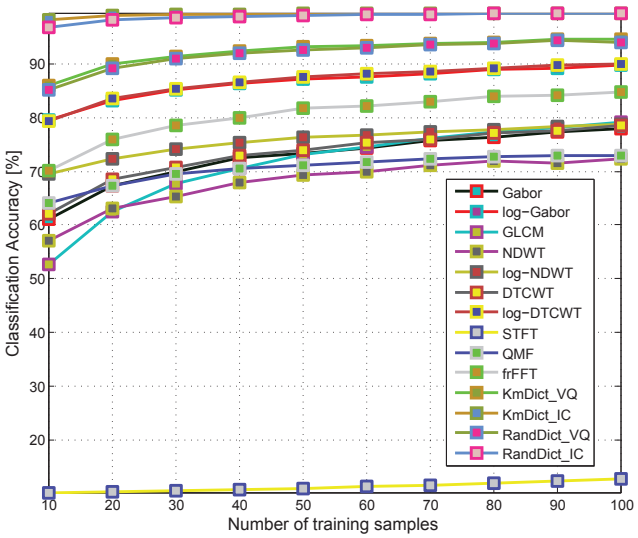


Figure 5.24: Comparison of BoW method with state-of-the-art methods.

methods significantly reduce both the computational effort and the memory requirements. Thus, our method is applicable and scalable to large databases. An extensive experimental evaluation has been performed and compared with other feature extraction methods. It is demonstrated that our feature extraction method is quite competitive and can achieve rather promising performance figures for SAR image classification. Therefore, vector quantization using vectorized patches and random dictionary is a very practical choice for real applications.

Although many feature encoding methods have been proposed, we do not see much improvement compared with vector quantization and in some cases they perform even worse than vector quantization. Based on the observation that all feature coding methods encode local features independently, we proposed a new incremental feature learning method for SAR image classification, which achieves significantly better results than current state-of-the-art methods. Incremental feature coding encodes the local descriptor accumulatively based on the previous images that have been analyzed. Therefore, it can learn a very discriminative feature representation.

5.6 The Bag-of-Spatial-Temporal-Words (BoSTW) Method

In this section, temporal image classification is carried out using the BoW method and the conclusions drawn from the previous section. In the previous section, the BoW features are learned to represent a single image. For temporal sequences of SAR images, the BoW features should be generated using low level features extracted from a compact volume neighborhood. Therefore, the temporal window size and its shift are two important parameters, which need to be investigated.

5.6.1 Extension to the Temporal Domain

The most important difference between a single image and a sequence of multi-temporal SAR images, is that multi-temporal SAR images have time related characteristics, which could be acquired at different times. There are two important objectives in analyzing multi-temporal SAR images; one is to look for temporal patterns that are similar without considering the physical scene

classes. In practice, different scene classes could have similar temporal patterns; for instance, in most cases, urban areas are quite stable, which means there is no obvious temporal pattern. In other areas like mountains, there is also not much temporal difference over a short time period. Therefore, mountains have the same temporal evolution pattern as urban areas. The second objective is to discriminate not only the temporal evolution patterns, but also the scene classes. In this case, we have much many classes. Consequently, we need a really large database and a lot of training samples from all classes to learn an efficient classifier. In this section, we focus on the second objective.

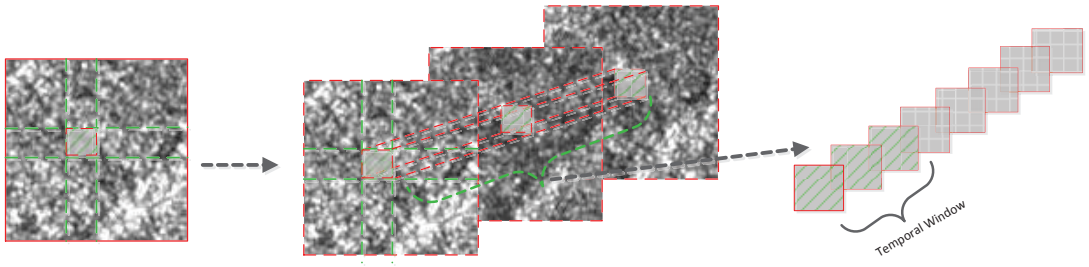


Figure 5.25: BoW model for temporal SAR images.

We extend the BoW method to multi-temporal SAR images, as shown in Fig.5.25. The SAR image sequence is split into small patches along the spatial and the temporal axis. All the pixel values in a compact 3D neighborhood are used as low level features. Thus, a SAR image sequence can be represented by the BoW features presented previously. Actually, any features that can be extracted from a compact 3D neighborhood can be used as low level features to generate the BoW feature vector. For example, a similarity matrix can be computed as a characterization of the temporal patterns, followed by a clustering of the symmetric similarity matrices to learn a dictionary. Since a 3×3 compact patch can give very good outcomes as concluded before, we use it in the spatial domain for temporal image classification. Therefore, the low level features are a concatenation of all 3×3 patches located in the temporal window.

5.6.2 Temporal Window Size

The temporal window size and its shift along the temporal axis are two key parameters in low level feature extraction. The image patch sequence within the temporal interval denotes the basic elements in the BoW model. The corresponding codebook consists of a number of representative temporal patterns. The temporal window size represents the length of the basic temporal patterns. With a short temporal window, we can find abrupt changes by change detection techniques. In this case, the SAR image sequence is represented as a collection of changing patterns, which can be applied to retrieve all similar changes in a SAR image sequence. Therefore, with a short temporal window size, we can find specific changes which happen at different times. On the other hand, with a large temporal window size, it is possible to find similar evolution patterns in the temporal domain. However, it is much more difficult to identify how many classes and which classes are contained in multi-temporal SAR images. Selecting an appropriate temporal window size becomes an important issue.

5.6.3 Evaluation and Discussion

In this section, BoSTW for multi-temporal SAR image classification is evaluated. A multi-temporal SAR image database consisting of 8 Radiometrically Enhanced (RE) and Geocoded

Ellipsoid Corrected (GEC) TerraSAR-X images, acquired before and after the Japanese tsunami disaster in 2011, were selected and used for testing. The pixel spacing of these images is about 2.9 m. Each image is cut into patches of 100×100 pixels, and 2305 patches from 13 classes are collected based on an active learning system Cui *et al.* [2013b]. Example images from all 13 classes are shown in Fig. 5.26. To highlight the temporal patterns, sequences of 3 successive images are visualized as a RGB color images, which is going to be presented in section 6.5 as a contribution to SAR ITS visualization. Thus, in each sequence, there are 6 RGB images. Due to the disaster effects, there are many real temporal patterns.

Since vectorized patches perform quite well, as presented previously, we use in the evaluation the concatenation of 3×3 vectorized patches from a fixed temporal window as low level features, and investigate the effect of the temporal window size on the accuracy of temporal SAR image classification. To validate the application of BoSTW for multi-temporal SAR image classification, in a second evaluation we compare the BoSTW feature extraction with the concatenation of other feature vectors.

Table 5.2: Performance investigation of the temporal window size.

Feature Encoding	Temp. Win. Size	100	200	300	400	500
VQ	1	78.82	80.51	80.88	81.81	81.69
	2	82.37	83.20	83.78	83.86	84.68
	3	82.40	83.36	83.44	83.66	84.33
	4	82.44	82.80	83.17	83.33	83.70
	5	82.19	82.72	82.59	82.90	83.23
	6	82.18	82.10	81.80	82.06	81.94
	7	81.05	81.48	81.59	81.66	81.60
	8	80.48	80.64	80.64	79.89	79.58
IC	1	98.68	98.67	98.27	98.29	98.65
	2	98.40	98.53	98.36	98.16	98.48
	3	98.49	98.38	98.40	98.13	98.28
	4	98.53	98.69	98.41	98.53	98.63
	5	98.64	98.34	98.27	98.56	98.57
	6	98.64	98.52	98.43	98.52	98.43
	7	98.21	98.56	98.46	98.46	98.71
	8	98.10	98.10	98.44	98.45	98.51

Table.5.2 shows the influence of the temporal window size on the classification accuracy using vectorized patches with two feature encoding methods. In our case, as there are many more temporal patterns, the dictionary has to be large enough. From the vector quantization results, we can see that a medium size temporal window should be employed. It appears that both too short and too long temporal windows will decrease the accuracy. The reason could be the undersampling of the temporal signals when we have too few samples on the temporal axis. A short temporal window cannot capture all temporal patterns because the BoW model consists of a collection of static images without any information about the temporal aspect. On the other hand, the temporal patterns may not be captured with a too long temporal window because the temporal variation becomes weak in the case of undersampling of the temporal signal. Thus, a temporal window of medimum size should be employed to capture the temporal patterns, but not necessarily a maximum size window.

To demonstrate the advantages of the BoSTW method for multi-temporal SAR image clas-

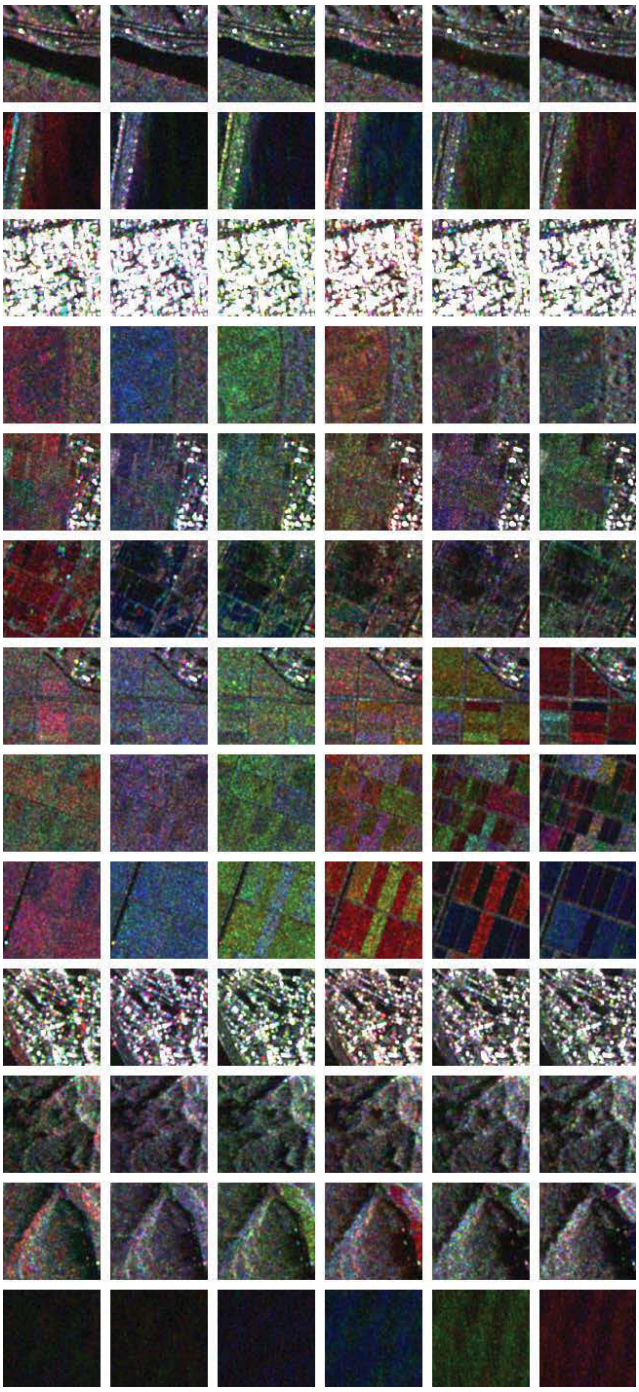


Figure 5.26: Example images of SAR ITS classes used for evaluation.

sification, we used Gabor features, NDWT features, DTCWT features, frFFT features and the corresponding log versions. The involved parameters are the same as in section 5.5.8. The only difference from section 5.5.8 is that the temporal features are just a concatenation of the features extracted from each image. A performance comparison of the selected methods for

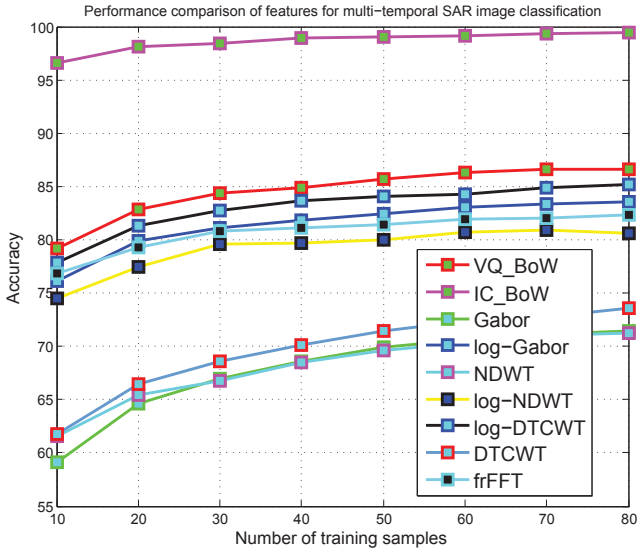


Figure 5.27: Performance comparison of feature extraction methods for multi-temporal SAR image classification.

multi-temporal SAR image classification is shown in Fig. 5.27. Similar to our previous results, incremental encoding achieves the highest accuracy. Vector quantization ranks second, which is still better than other methods. In this test, the log versions of the extracted texture features perform only slightly worse than BoSTW because the images in the sequences contain typical stationary textures, which are quite different from the database used previously.

Chapter 6

Cascaded Active Learning for Spatial and Temporal SAR Image Information Mining

With increasing satellite image acquisition rates and image resolution, both image content and temporal patterns are becoming quite diverse, which makes an automatic and fast interpretation of large data volumes a highly demanding technique. When we have large databases of satellite images aimed at real applications, it is not trivial to know which patterns and how many of them exist in the given images. This is the most important and practical issue, and makes the generation of ground truth data very difficult. Therefore, the development of methods and systems for exploring large scale databases is of significant practical importance.

In this chapter, a cascaded active learning approach relying on a coarse-to-fine strategy for spatial and temporal SAR image information mining is developed, which allows fast indexing and the discovery of hidden spatial and temporal patterns in multi-temporal SAR images. An overview of the cascaded active learning approach is presented in section 6.1. The next two sections (i.e., 6.2 and 6.3) will present our basic SVM and active learning algorithms. In addition, a comparison of sample selection strategies in active learning is performed. The implementation of our method is presented in section 6.6. It involves two practical issues, which are the visualization of multi-temporal SAR images and sample propagation between levels. Multiple instance learning is applied in section 6.4 to automatically select the most informative samples for a new level and a color animation representation is introduced in section 6.5 to avoid information distortion. At the end, an evaluation which concludes this chapter is given in section 6.7.

6.1 Overview of Cascaded Active Learning

The cascaded active learning approach is developed to manage the increasing volumes of satellite image archives; it allows fast annotation and the discovery of hidden patterns in multi-temporal SAR images. The motivation of the system is to disregard as many irrelevant patches [1] as possible at a coarse level and to focus the learning and the computational effort on the relevant patches at a higher level. At each level, a relevance feedback based on active learning is performed, in which the most informative samples to the classifier are selected and manually labeled. The framework of the cascaded active learning approach is shown in Fig. 6.1. It is mainly composed of

[1] In this thesis, by relevant patches, we mean the patches that contain at least a small fraction of the target class. Thereby, irrelevant patches are those that do not comprise any part of the target class.

three components, which are *feature extraction*, *visualization*, and *cascaded active learning*. Image representation and feature extraction are the fundamental components of almost all applications. In our approach, a hierarchical structure for image representation is used, which can significantly speed up the learning by dropping all irrelevant parts. Feature extraction, especially BoW feature extraction, has been presented in detail in the previous chapter. Visualization is an important part, especially for multi-temporal SAR images, and, as a rule, involves dimension reduction. However, any dimension reduction method may distort the image content and lead to a loss of information. To keep all the information visible to expert users, and to highlight the content variation, color animation is used. The learning part is at the heart of the approach, which includes SVM active learning and multiple instance learning. Relevance feedback based on SVM active learning is carried out at each level to eliminate all irrelevant patches. The core component of active learning is the strategy to select patches for feedback. In addition, the method has to be fast enough such that the users can get feedback immediately after selecting training patches. After learning on one level, all the relevant patches are moved up to a finer level. At the same time, multiple instance learning is used to automatically propagate the training samples to the higher levels, which can further reduce the manual effort. This cascaded active learning is repeated until a very detailed level. Therefore, with this approach, a coarse-to-fine annotation can be obtained.

The overall procedure of our cascaded active learning is presented in the middle part in Fig. 6.1. Starting from the first level, all patches are classified either as positive or negative. However, positive patches probably contain some other classes, as shown in the classification results at the first level (the two patches between *water* and *beach*). These positive patches are classified further at the next level to eliminate the irrelevant parts. Nevertheless, training samples are not available at the finer level, and have to be inferred from the samples at the previous level. As each training sample is considered as a bag containing sub-patches as instances, the optimal positive sub-patches can be learned by multiple instance learning. Using the learned optimal sub-patches as training samples, all the sub-patches of the positive patches at the previous level are classified again, resulting in a better annotation.

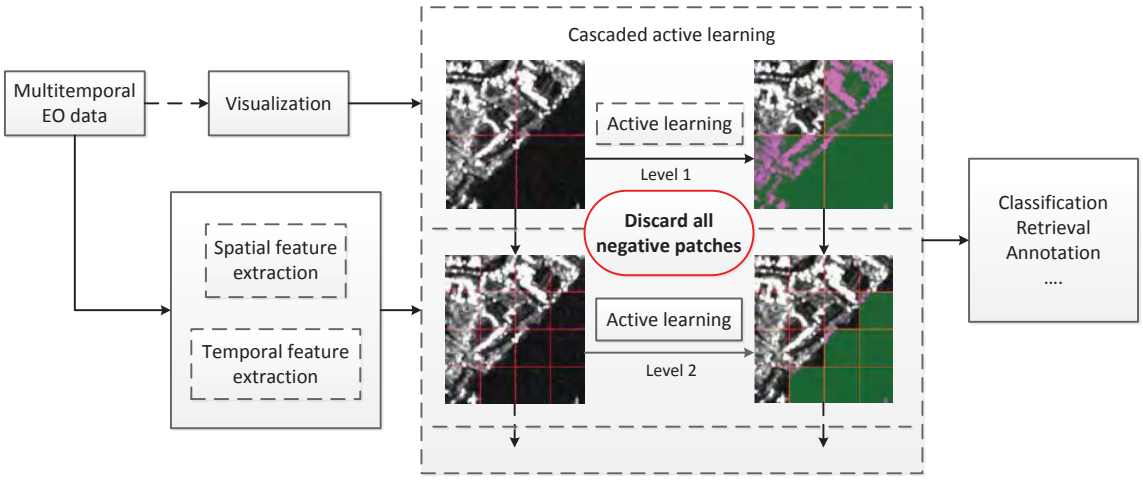
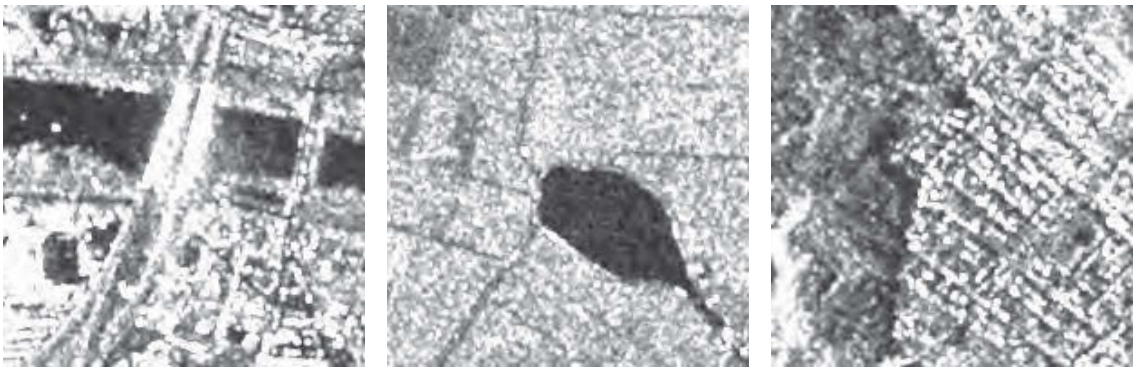


Figure 6.1: Overview of cascaded active learning for multi-temporal SAR image information mining.

6.1.1 Image Representation and Feature Extraction

Throughout this thesis, we work at the patch level. Nevertheless, in most cases, a patch probably contains several classes, as shown in Fig. 6.2. The first patch contains a river, a bridge and some buildings. In the second patch, there is a small lake in the middle of the agricultural field. In the third patch, there are some buildings at the foot of the mountain. As can be seen, each patch has a local context corresponding to a specific configuration of several classes. This context is an important part of image analysis, which should be taken into account during the learning process. Without this context, the performance of any recognition and classification can be degraded. To annotate the classes within the patches, we need to cut the patches into smaller sub-patches but preserve the local context when performing classification, which results in a hierarchical patch representation of the image as shown in Fig. 6.3.



(a) (b) (c)

Figure 6.2: Example patches that include several classes: (a) a patch with a river, a bridge and some buildings; (b) a patch with a lake in the middle of the agricultural field; (c) a patch with some buildings at the foot of the mountain.

Hierarchical image representations have been widely used in image interpretation. Their merit is that they can represent a scene in a coarse-to-fine manner, which is especially useful for object recognition. During cascaded learning, we rely on this advantage to discard all irrelevant patches at the coarse level such that the computational effort can be significantly reduced, and we can fulfil the strong speed requirements of our approach. After dropping all the irrelevant patches, the classification is applied only to the potential target patches. As an image is represented using a hierarchy, feature extraction is applied independently to all its levels. The feature vector being used in this approach is the BoSTW feature vector presented in the previous chapter, which has been demonstrated to be more efficient for image classification than conventional texture features. It represents a generalization of BoW feature to temporal space by concatenating the low level features in a compact 3×3 window. An obvious benefit of learning at different levels is that we can track the feature space at different levels, which is important for information mining. In addition, as classes are defined independently at each level, and a taxonomy can be obtained as a semantic hierarchy which can serve as a basis for other applications playing an increasingly important role in image understanding.

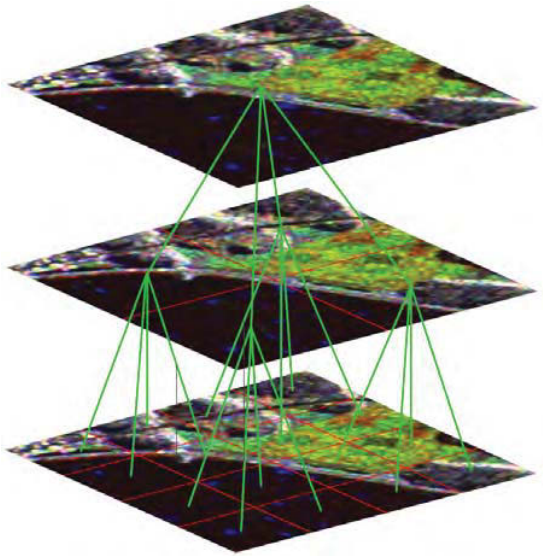


Figure 6.3: Hierarchical image representation.

6.1.2 Learning Algorithms

In this approach, three learning paradigms are employed, which are *SVM*, *active learning*, and *multiple instance learning (MIL)* Dietterich *et al.* [1997]. As one of the best supervised classifiers, SVM is used as a basic element in our cascaded active learning method. The reason why we chose SVM is that it can efficiently manage high dimensional feature vectors while preserving a high accuracy. In addition, it has a good generalization capability, even for nonlinearly separable cases. To learn a good SVM, sufficient training samples are required, which is not trivial for practical cases. To solve this problem, both semi-supervised learning and active learning have received a considerable amount of interest. In semi-supervised learning, both labeled and unlabeled data are used for learning, which make use of two main assumptions, namely the smoothness and the manifold assumption. The smoothness assumption states that feature vectors that are close to each other are more likely to share a label, while the manifold assumption supposes that the data lie approximately on a manifold of much lower dimension than the input space. However, due to the consideration of unlabeled data, the computational burden is always a problem, which makes it infeasible to apply the concept in practical working systems.

In order to fulfil the speed requirement, we apply active learning (also called "query learning" or "optimal experimental design" in the statistics literature) in our approach to incrementally obtain reliable samples. The key assumption in active learning is that if it is allowed to query data it will perform better with less training (see Settles [2009]). In practical applications, the labels of the data are very difficult to obtain, the task is either time-consuming or computationally expensive. Active learning tries to overcome this bottleneck by querying important unlabeled data and asking expert users for manual labeling. Therefore, active learning can achieve better accuracy while using few important labeled instances, thus, minimizing the cost of collecting labels and reducing the manual effort of annotation. Obviously, two important components in active learning are classifier training, using the already labeled images, and sample selection, which selects the most informative samples for manual labeling. These two components work alternatively, which can significantly reduce human labeling effort and achieve better performance for image indexing.

As presented in section 6.1.1, the learning is performed on a hierarchy. The training samples manually selected by expert users are only available at the first level. These samples cannot be used for higher levels because they probably contain more than one class as shown in Fig. 6.2. Thus, we need a solution to find training samples for the higher levels. Two possible solutions exist. One is to ask expert users to select manually training samples again for a new level, which would not be acceptable to them. The second solution is to automatically infer the samples for a new level from the previous level, thereby further reducing the manual labor effort. This would be more preferable than the first solution from the expert users' point of view. Theoretically in machine learning, MIL is a good candidate solution to this problem, which solves this problem by considering each patch as a bag, with sub-patches at the next level as data points in a bag. In contrast to conventional supervised learning, the training data in MIL is given in the form of pairs of bag and label rather than pairs of instance and label. The basic assumption is that there is at least one positive instance in each positive bag and all instances in a negative bag are negative. Thus, the difficulty in MIL stems from the label ambiguity, which gives an integer programming problem. A heuristic method is used to solve the integer programming problem. In the context of MIL, it is assumed that the negative instances in the positive bags are similar to the negative instances in negative bags. Therefore, the negative samples will be replaced with the neighbors of all the patches that are classified as being positive.

6.1.3 Cascaded Classifier

In most EO applications, every class covers only a certain part in a large image. Thus, it is not really necessary to process all the patches because only some of them are relevant to the target class. In addition, because of the complexity of learning a strong classifier, it is not necessary to use a strong classifier while a weak classifier can complete the task, thereby saving computational effort remarkably. In contrast, a strong classifier should be used to focus the learning on the relevant patches such that the accuracy can be improved. Thus, the learning method should discard all irrelevant patches as early as possible and focus the training and learning on the relevant patches. In this way, the computational burden can be significantly decreased and the target class can be discovered quickly, which is extremely important in EO applications because the data volume is much larger than in other fields. As we work on patches, one patch is probably relevant to more than one class. To obtain a pure class and reject irrelevant parts, the patches should be further sub-divided, which leads to a multi-scale hierarchical structure as shown in Fig. 6.3.

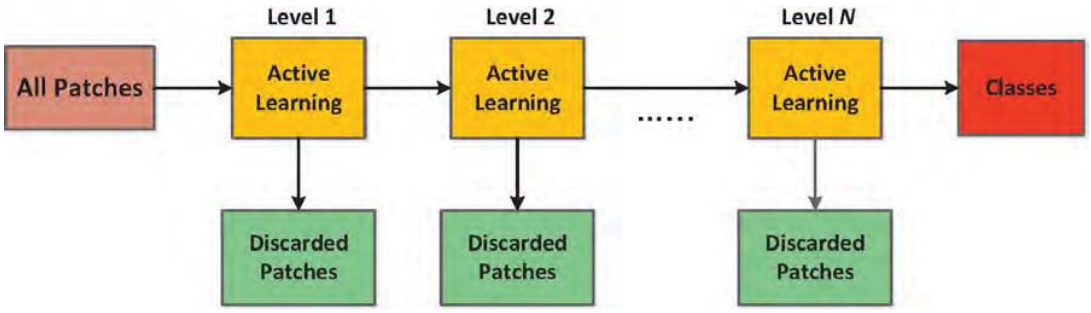


Figure 6.4: Cascaded classifier.

Based on this motivation, a cascaded classifier shown in Fig. 6.4 is employed for multi-temporal SAR image information mining with the aim to reduce the amount of data to be

processed thus speeding up the learning remarkably. Various cascaded classifiers have been developed and applied to object detection in computer vision, especially for face detection by Viola & Jones [2004] and Wu *et al.* [2008]. They have achieved remarkable efficiency in terms of both accuracy and computational effort. However, in the field of EO, cascaded classifiers have not yet been exploited for image information mining, which is our focus in this chapter. A cascaded classifier is a concatenation of several classifiers learned at different stages using all information collected from the output of a given classifier as additional information for the next classifier in the cascaded structure. Thus, if a patch has been rejected by one classifier, no further training and classification will be performed on its sub-patches in the next stage. In this way, the amount of patches that need to be processed can be significantly reduced while preserving a high accuracy. It is worth noting the difference compared to ensemble and boost classifiers. A cascaded classifier is a multistage classifier while ensemble and boost classifiers are some combinations of several classifiers. With a cascaded classifier, hidden patterns can be discovered quickly and accurately because a large fraction of patches will be rejected at each level and the computation focuses only on the relevant patches. At every level, a classifier is learned incrementally through active learning and all the negative patches are rejected while keeping only the positive ones. Then all positive patches are sub-divided and all the sub-patches are imported into the next level. The classification at the next level focuses only on these sub-patches. Between levels, positive samples are propagated through MIL, while negative patches are replaced with the neighbors of all the patches that are classified as being positive.

6.2 Support Vector Machine (SVM)

6.2.1 Preliminaries

Due to the speed requirement for the system response, we selected a SVM classifier. Thus, we first briefly review the SVM theory. We first consider the linear case of a two-class classification. Given N training samples $X = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, with $x_i \in R^M, y_i \in \{-1, +1\}$, the objective of SVM is to find a hyperplane $w^T x + b = 0$ with a normal vector w and bias b that can separate the two classes with a maximum margin as shown in Fig. 6.5(a), which underlies the following constraints

$$\begin{aligned} w^T x_i + b > 0 \quad \text{for} \quad y_i = 1 \\ w^T x_i + b < 0 \quad \text{for} \quad y_i = -1 \end{aligned} \tag{6.1}$$

To define the concept of margin, the separating hyperplane can be shifted up and down by the same distance $c = 1$ (c can be any constant value, dividing both sides of the constraints by c would give the same inequality constraints as Eq. (6.2.), resulting in the following constraints

$$y_i(w^T x_i + b) \geq 1 \quad \text{for} \quad i = 1, ..., N \tag{6.2}$$

The distance between the two parallel hyperplanes $\frac{2}{\|w\|}$ is considered as the margin, which should be maximized such that it has the best generalization capability. Thus, the two class linear SVM can then be formulated as the following quadratic optimization problem, which is the primal form of a hard margin SVM.

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ \text{s.t.} \quad & y_i(w^T x_i + b) \geq 1 \quad \forall i = 1, ..., N \end{aligned} \tag{6.3}$$

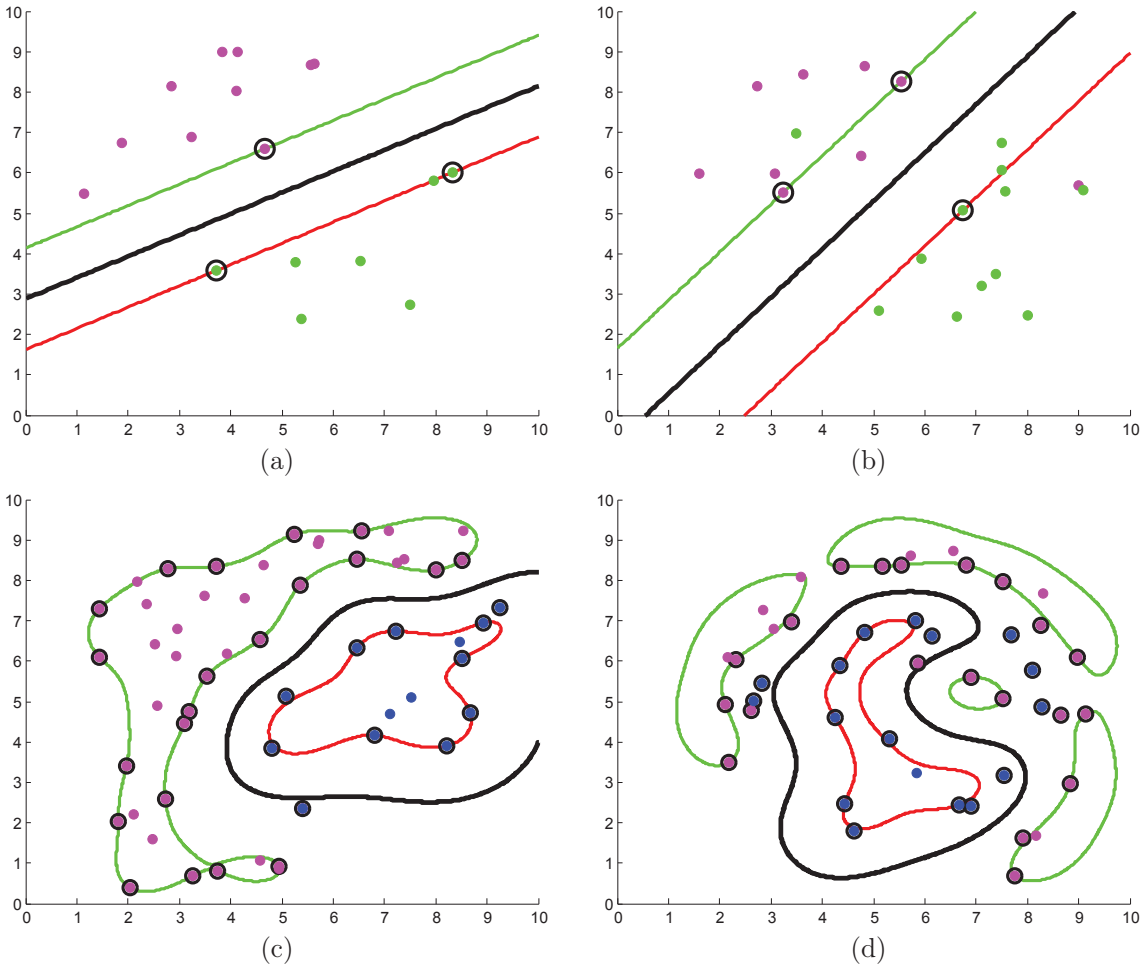


Figure 6.5: Optimal separating hyperplanes: (a) Hard margin SVM; (b) Soft margin SVM; (c) Nonlinear SVM in the separable case; (d) Nonlinear SVM in the non-separable case.

However, a hard margin SVM can only manage the linearly separable case, which is quite limited in practical applications. To overcome this drawback, and minimize misclassification of some data points between the two parallel hyperplanes, slack variables ξ_i are introduced to relax the hard constraints. Misclassifications and the margin shall be minimized; this leads to the primal form of a soft SVM in Eq. (6.4), with C being a regularization parameter controlling the trade-off between goodness-of-fit and generalization capability. An example is shown in Fig. 6.5(b), where the support vectors are marked by black circles.

$$\begin{aligned} &\min_{w,b,\xi_i} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \\ &\text{s.t.} \quad y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \forall i = 1, ..., N \end{aligned} \tag{6.4}$$

A higher value of C would penalize the error more, which allows less misclassification, leading to a model better fitting the data, but with a risk of overfitting. On the contrary, a smaller C would give an underfitting model, which allows more misclassification.

The optimization in Eq. (6.4) is a quadratic programming problem, which can be solved by convex optimization software, like the CVX package [1]. To solve the quadratic optimization problem in Eq. (6.4), we convert it to an unconstrained optimization problem by introducing a Lagrange multiplier for each inequality constraint, and the new objective function becomes

$$Q(w,b,\xi,\alpha,\beta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i\Big(y_i(w^T x_i + b) - 1 + \xi_i\Big) - \sum_{i=1}^{N}\beta_i\xi_i \tag{6.5}$$

An optimal solution has to satisfy the Karush-Kuhn-Tucker (KKT) condition (see Kuhn & Tucker [1950]), Taking the derivative with respect to the unknown parameters gives the following KKT condition.

$$\frac{\partial Q(w,b,\xi,\alpha,\beta)}{\partial w} = w - \sum_{i=1}^{N}\alpha_i y_i x_i = 0 \tag{6.6}$$

$$\frac{\partial Q(w,b,\xi,\alpha,\beta)}{\partial b} = \sum_{i=1}^{N}\alpha_i y_i = 0 \tag{6.7}$$

$$\frac{\partial Q(w,b,\xi,\alpha,\beta)}{\partial \xi} = C - \alpha - \beta = 0 \tag{6.8}$$

$$\alpha_i\Big(y_i(w^T x_i + b) - 1 + \xi_i\Big) = 0, \quad \text{for} \quad i = 1,...N, \tag{6.9}$$

$$\beta_i\xi_i = 0, \quad \alpha_i \geq 0, \quad \beta_i \geq 0, \quad \xi_i \geq 0, \quad \text{for} \quad i = 1,...N. \tag{6.10}$$

From Eq. (6.8), (6.9), and (6.10), there are three cases for α_i (Abe [2005])

1. if $\alpha_i = 0$, then $\xi_i = 0$. In this case, x_i is correctly classified.
2. if $0 < \alpha_i < C$, then $y_i(w^T x_i + b) = 1$ and $\xi_i = 0$. The data points meeting this condition are called unbounded support vectors and determine the decision surface. The other data points do not play a role in determining the classifier.
3. if $\alpha_i = C$, then $y_i(w^T x_i + b) - 1 + \xi_i = 0$ and $\xi_i > 0$. In this case, these data points are located between the two parallel hyperplanes. If $0 \leq \xi_i < 1$, x_i is correctly classified, and if $\xi_i \geq 1$, x_i is misclassified.

To solve the optimization problem in Eq. (6.5), substituting Eqs. (6.6) to (6.8) into Eq. (6.5) gives the dual form objective function of a hard margin SVM

$$Q(\alpha) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j x_i^T x_j \quad \text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1,...,N. \tag{6.11}$$

From the dual form given by Eq. (6.11), the number of variables to be optimized is the same as the number of training samples, and the objective function depends only on the dot product of the data points. These are the main advantages of the dual form over the primal form, where the number of variables to be optimized is the same as the dimension of the feature vector; however, it is hard to extend it to nonlinearly separable cases. Thus, if the dimension of the feature vector is high and only few training samples are available, the dual form is preferable to the primal form.

[1] http://cvxr.com/cvx/

Although a soft margin SVM classifies nonlinearly separable data points by allowing a certain number of them being misclassified, it cannot manage a nonlinearly separable feature space. To overcome this drawback, we introduce the kernel trick by mapping $\Phi : R^N \rightarrow R^M, M \geq N$ the data points to a high dimensional separable space $\mathcal{F} = \{\phi(x) \mid x \in R^N\}$ based on the fact that the dual form objective function depends only on the dot product of the feature vector. Using the mapped data points, a linear decision surface $w^T\phi(x) + b = 0$ in the high dimensional space can be learned, and the corresponding dual form optimization is given in Eq. (6.12).

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \phi(x_i)^T \phi(x_j) \quad \text{s.t.} \quad \sum_{i=1}^{N} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1, ..., N. \tag{6.12}$$

The only difference of Eq. (6.12) from the linear case of Eq. (6.11) is that the dot product between the feature vector is replaced with the dot product in the high dimensional feature space after mapping. However, it is difficult to identify what feature mapping can separate a nonlinear feature space. Nevertheless, the most important achievement in the theory of SVM is that the dot product after feature mapping is equivalent to a kernel function under some conditions based on the Hilbert-Schmidt theory $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$. This is the so-called kernel trick. Therefore, the difficulty to identify a feature mapping function $\phi(x)$ is avoided by a kernel function. The optimization in dual form becomes Eq. (6.13) and depends only on the Gram matrix $\mathbf{K}$ of the data points with each entry representing the similarity of two feature vectors in terms of a kernel function $K_{i,j} = K(x_i, x_j)$.

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad \text{s.t.} \quad \sum_{i=1}^{N} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1, ..., N. \tag{6.13}$$

The corresponding decision function of a test feature vector is

$$f(x) = \sum_{i \in S} \alpha_i y_i K(x_i, x) + b, \tag{6.14}$$

where S is the set of support vectors. One important quantity in using a nonlinear SVM is the distance to the decision surface shown in Eq. (6.15), which is usually used as a informativeness measure for active learning. An example of the distance to the decision surface is shown in Fig. 6.6(a).

$$D(x) = \frac{\sum_{i \in S} \alpha_i y_i K(x_i, x) + b}{\|w\|} \tag{6.15}$$

The complexity of learning a SVM classification depends on the dimension of the Gram matrix, that is the number of training samples. The remaining problem is to choose an efficient kernel function. In most image processing applications, histogram based feature representations of images are becoming the state-of-the-art method. As verified empirically, the histogram intersection kernel and the Chi square kernel, shown in Eq. (6.16), perform better than Euclidian distance. In addition, they are additive kernels (Maji *et al.* [2013]), which can be computed quickly. In the following, we use a Chi-square kernel.

$$K_{\chi^2}(x, y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{x_i + y_i} \quad K_{HIK}(x, y) = \sum_{i=1}^{n} \min(x_i, y_i) \tag{6.16}$$

6.2.2 Probabilistic SVM Output

One important drawback of SVM is that it does not give a probabilistic interpretation of the decision value. In some applications, a probabilistic output of the form $p(y|x)$ is required. a sigmoid function was proposed by Platt [2000] to transform the SVM decision value to a posterior probability $p(y = 1|x)$ by Eq. (6.17), thus $p(y = -1|x) = 1 - p(y = 1|x)$.

$$p(y = 1|x) = \frac{1}{1 + \exp\left(Af(x) + B\right)} \tag{6.17}$$

Assume that the label y of the feature vector follows a Bernoulli distribution with a parameter $p(y = 1|x)$ such that $p(y = 1|x) = 1$ for a positive instance and $p(y = -1|x) = 1$ for a negative instance. Then the two parameters A and B can be estimated by a maximum likelihood estimator. The negative log-likelihood function is given by Eq. (6.18).

$$L(A, B) = \sum_{i=1}^{N} \frac{1 - y_i}{2}\Big(Af(x_i) + B\Big) - \log\Big(1 + \exp(Af(x_i) + B)\Big) \tag{6.18}$$

An example of the probabilistic output is shown in Fig. 6.6(b). The probabilistic output can also be used as a informativeness measure for active learning

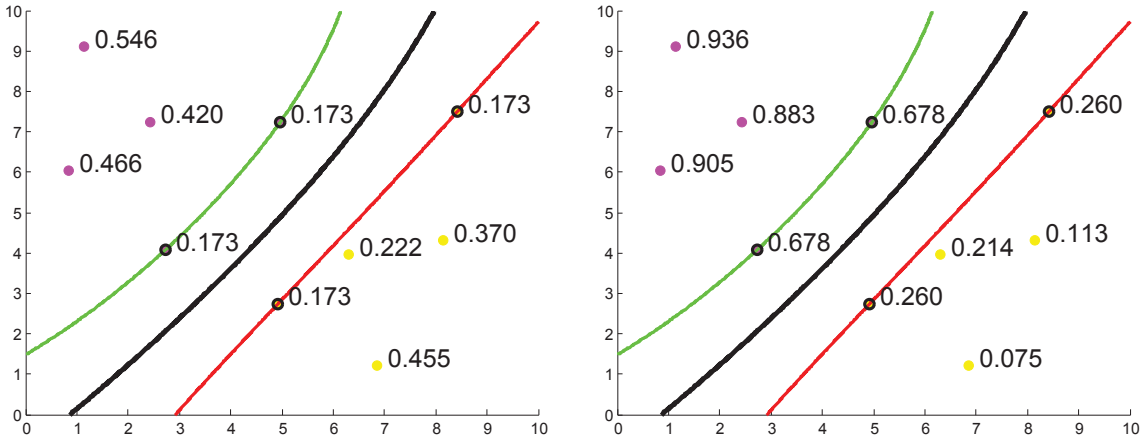


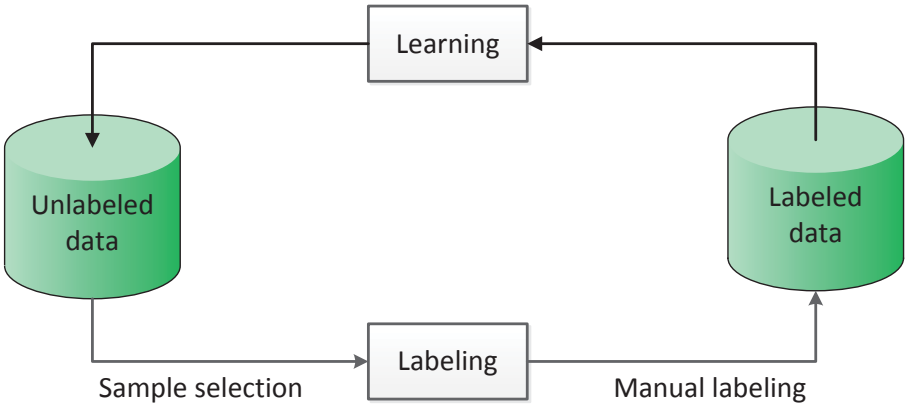
Figure 6.6: Optimal separating hyperplanes: (a) Distance to the separating hyperplane; (b) Probabilistic SVM output.

6.3 SVM-based Active Learning

In practice, it is usually costly to obtain labeled data due to limited financial and human resources and the advantage of building a compact yet sufficient training dataset has been realized. Starting from a limited number of labeled data, active learning methods select the most informative samples to speed up the convergence of learning and to reduce the manual effort of labeling. In active learning, our aim is to query the important instances in the feature space to maximize the accuracy and shorten the learning time. Actually, many labeled instances in a given training dataset are not helpful to learn an efficient classifier. Instead, they increase the computational

complexity of learning. To eliminate those redundant samples thus keep only the most informative ones is the main objective of active learning. The two core components in active learning are the selection strategy and model learning, which are repeated iteratively until convergence. In this chapter, a SVM is adopted as the component for model learning.

The overall framework of active learning is shown in Fig. 6.7. At the beginning of the procedure, there are only a few labeled instances available, and a coarse classifier is learnt. After that, the two components are iteratively performed until the classification result is satisfactory. In each iteration, the most informative samples from the unlabeled instances in the pool are selected and labeled manually by expert users to improve the current classifier. These newly labeled instances are added to the training set, and a new classifier is learnt based on the increased number of labeled instances. There are many different active learning paradigms, such as membership query learning, stream-based active learning, and pool-based active learning (Settles [2009]). In this chapter, we focus on pool-based active learning, where a large pool of unlabeled data and only a small set of labeled data are available. In each iteration, queries are selected from the pool based on an informativeness measure. This sample selection strategy plays a crucial role in active learning. In the following sections, we review various sample selection strategies.



(a)

Figure 6.7: Overall framework of active learning.

6.3.1 Version Space

Given a set of labeled instances $x_i, i = 1, 2, .., N$ and a kernel function $K(x, y)$, there are many hyperplanes $y = w^T\phi(x) + b$ in the transformed feature space $\mathcal{F}$ that can separate the actual classes. The set of all hyperplanes is called the version space (Tong [2001]). Formally, it is defined as

$$\mathcal{W} = \{w \mid \|w\| = 1, y_i(w^T\phi(x)) > 0, i = 1, 2, ..., N\} \tag{6.19}$$

From the linear equation $y = w^T\phi(x) + b$, a point and a hyperplane in the two spaces $\mathcal{F}$ and $\mathcal{W}$ are dual representations of each other. In other words, a point in one space corresponds to a hyperplane in the other space and vice versa. Specifically, given a new instance (x_i, y_i), any separating hyperplane in the version space has to satisfy the condition $y_i(w^T\phi(x_i) + b) > 0$. Instead of considering w as the normal vector of the hyperplane, the transformation $\phi(x_i)$ of x_i can also be viewed as the normal vector. Thus, in the version space, each labeled instance

defines a half space. If we normalize the entire version space to the unit hypersphere, the feasible version space is a connected segment on the surface of a hypersphere. The normal vector of the optimal SVM decision surface is the center of the largest hypersphere that can fit inside the current version space. The instances that correspond to these hyperplanes are the support vectors.

6.3.2 Sample Selection Strategies

Based on the notation of version space, the objective of active learning is to query the instances that can reduce the size of the version space as much as possible. To speed up the learning, the instance selected in each iteration should split the current version space into two equal parts. However, it is not practical to explicitly compute the sizes of the two parts given a new unlabeled instance which separates the version space into two parts. If we assume that the version space is symmetric, the normal vector w^* of the optimal decision surface is often roughly in the center of the version space. With this assumption, the instance that corresponds to the closest hyperplane to the optimal w^* in the version space should be able to separate the version space into two equal parts. Thus, the unlabeled instance from the pool that is the closest one to the current decision surface should be queried as the most informative one.

A toy example of active learning is shown in Fig. 6.8. Four classes are generated by the Gaussian mixture model with four components, where each component is assumed to belong to a class. Starting from the initial training set with one sample from each class, 16 informative samples that are close to the decision surface are added to the training set. Both the decision surfaces and the training samples in the first 8 iterations are shown in Fig. 6.8. In addition, the resulting classification accuracy vs. iteration is shown in Fig. 6.8(i). From this curve, we can see that the accuracy converges after 5 iterations. This sample selection strategy is called margin sampling or uncertainty sampling.

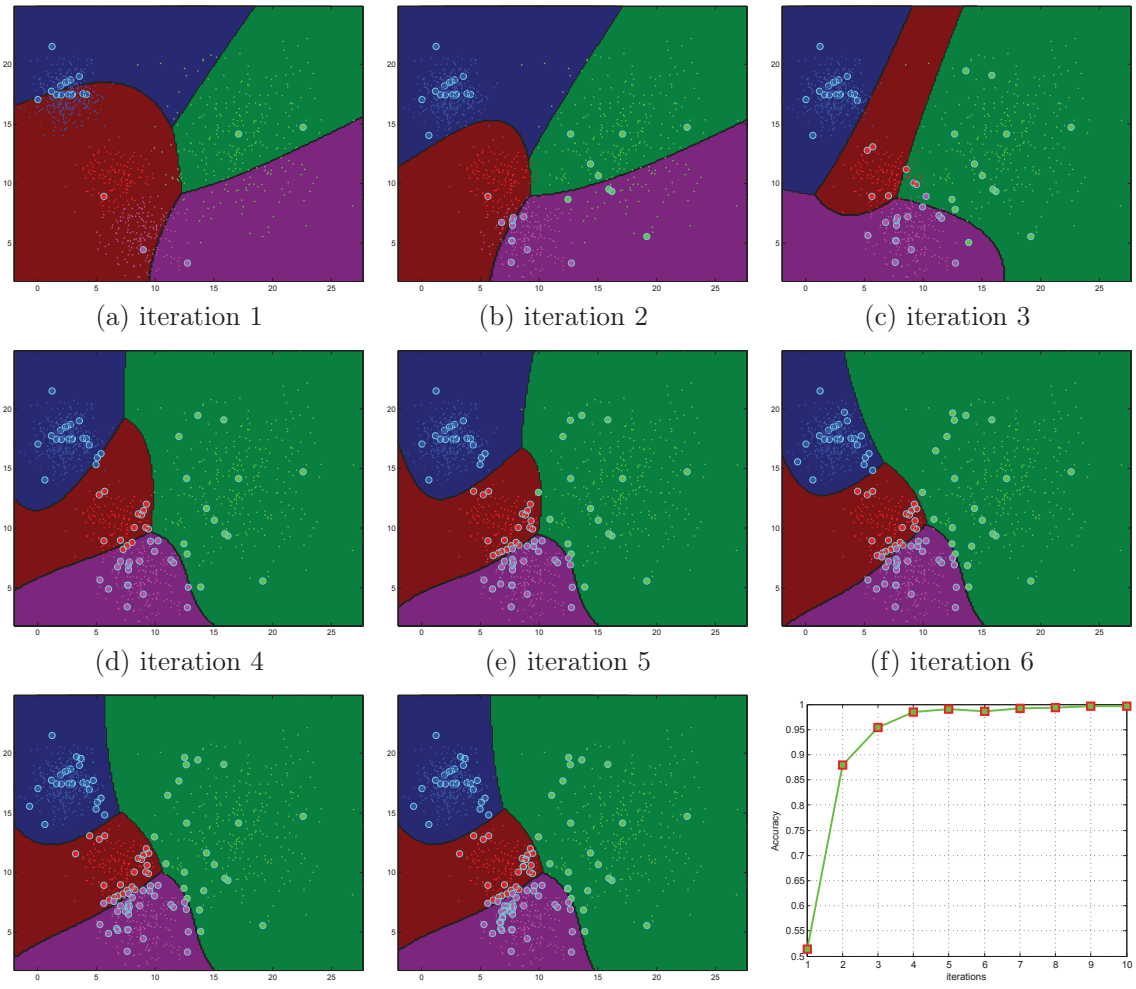
In addition to this strategy, there are some other methods for sample selection. Previous studies show that the selected samples should be diverse with respect to the currently labeled instances. The most popular approach is angular diversity between two samples, which is defined as

$$|\cos(< x_i, x_j >)| = \frac{|K(x_i, x_j)|}{\sqrt{K(x_i, x_i)K(x_j, x_j)}} \tag{6.20}$$

Based on the definition of angular diversity, the angle between an unlabeled and the current training samples is defined as the maximum angle from samples x to the labeled instances $\mathcal{L}$. Therefore, the diversity of a sample x in the pool is given by

$$\text{Div}(x) = 1 - \max_{x_i \in \mathcal{L}} \frac{|K(x, x_i)|}{\sqrt{(K(x, x)K(x_i, x_i))}} \tag{6.21}$$

Another often used method is density sampling. Density measures aim to select samples from dense unlabeled regions in the feature space based on the assumption that the samples in the dense regions are representative of the underlying distribution. Thus, they can add much more information to the learning model compared with samples in low density regions. Kernel Density Estimation (KDE), presented in section 4.1.6.2, is usually applied to density estimation. To reduce the computational effort, the density can be estimated using only the neighbors rather than all the samples.



(a) iteration 1 (b) iteration 2 (c) iteration 3

(d) iteration 4 (e) iteration 5 (f) iteration 6

(g) iteration 7 (h) iteration 8 (i) Accuracy vs. iterations

Figure 6.8: Evolution of the decision surface in SVM active learning.

6.3.3 Comparison and Discussion

To evaluate these sample selection strategies, the database for feature evaluation presented in the previous chapter is used. In total, there are 3434 images, which are partitioned randomly into three parts corresponding to training data, test data, and pool data with 200, 1144, and 1090 elements respectively. 20 active learning iterations are performed and 50 samples are selected and labeled in each iteration. Four different features have been used for evaluation, which are Gabor texture features and its log-version, BoW feature vectors using SRP Global and vectorized patches with local features from 3×3 patches together with vector quantization and incremental coding. For the sake of comparison, random sampling is presented as a baseline.

The classification accuracy vs. iteration is shown in Fig. 6.9. As the number of training samples increases, the accuracy becomes better and better. It can be clearly seen that margin sampling performs quite well compared with other sample selection methods for all four feature extraction methods. In some cases, the accuracy decreases locally for a few iterations because of the different initial training samples. In our case, the initial training samples are randomly

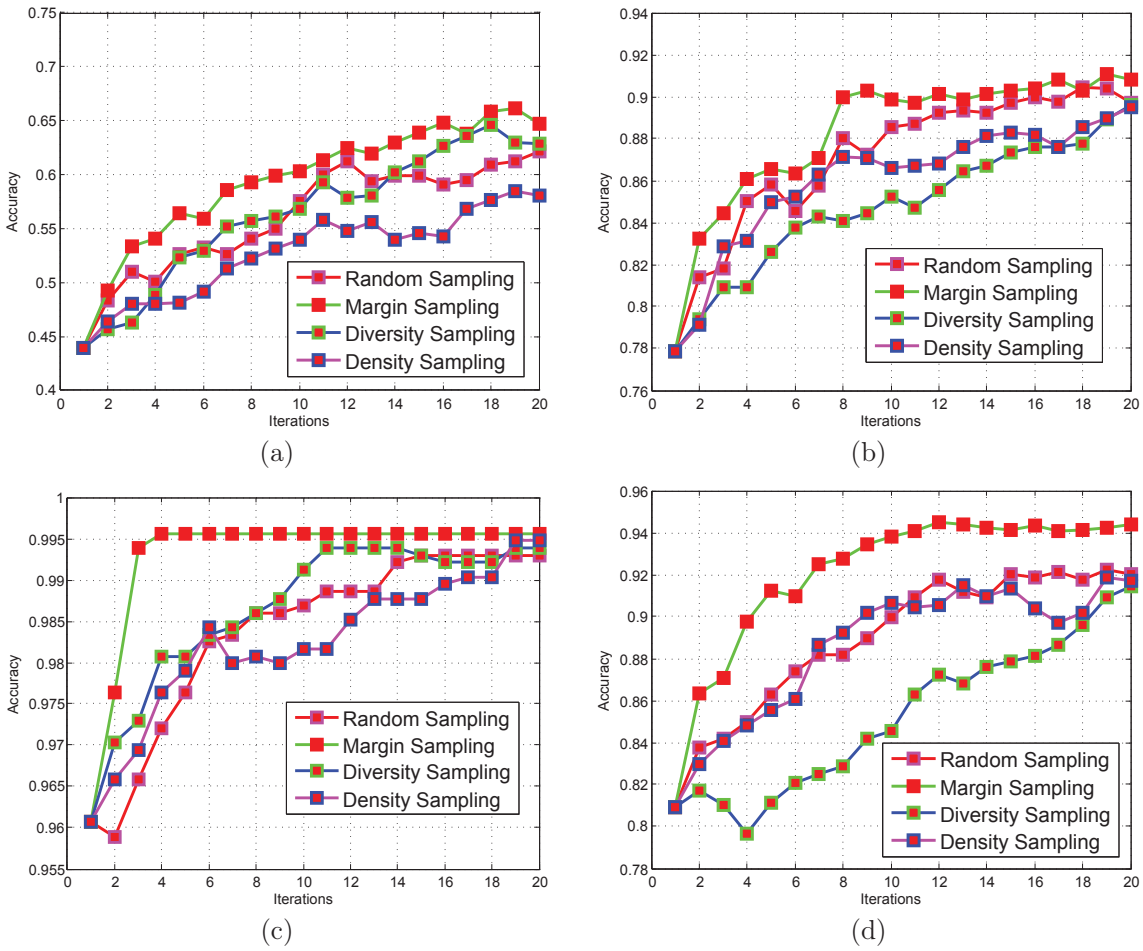


Figure 6.9: Comparison of different active learning methods using different features: (a) Gabor texture features; (b) Log-version of Gabor texture features; (c) BoW feature vectors using vectorized patches with local features from 3×3 patches and incremental coding with a dictionary size of 450 entries; (d) BoW feature vectors using SRP Global with local features from 3×3 patches and vector quantization with a dictionary size of 250 entries.

selected. But the overall trend is that the accuracy increases as more and more informative samples are collected for training. Another observation is that the convergence speed depends on the discrimination capability of the features. For less discriminative features, (e.g., Gabor features), all the four sample selection methods converge slowly. In contrast, for very discriminative features, (e.g., BoW feature vectors shown in Fig. 6.9(c)), margin sampling converges quite fast after only four iterations. Although density and diversity sampling were proposed for active learning, they perform equally or even worse than random sampling. This leads to the conclusion that the data points close to the decision surface are more important than the points in dense diverse regions. This conclusion is confirmed by the data point selection method proposed by Li & Maguire [2011]. In addition to the fast convergence of margin sampling, its computational complexity is much lower than that of the other methods, since the decision value is used directly as a measure of informativeness, which can significantly speed up the computation and thus becomes practical for a working system.

6.4 Multiple Instance Learning (MIL)

Another issue in the system is how to infer the training samples from the ones that had been selected or learned at the previous level. For negative patches, it is straightforward to use all the sub-patches as negative samples. However, for positive patches, it is not trivial to determine which sub-patch is positive because we know only that there is at least one positive instance, which means there are probably negative ones, too. Propagation of the training samples between two successive levels is approached by SVM multiple instance learning. A key assumption of MIL is that the negative instances in negative bags are similar to the negative instances in positive bags. In our context, the negative bags are selected by the user and they are very unlikely similar to the negative instances in the positive bags, thus this assumption may not hold anymore. Therefore, we replace the negative patches selected by the user with the spatially neighboring patches of positive bags at next levels. Then all these negative bags and positive ones are input to the MIL learning to identify the true positive instances.

In contrast to supervised learning, the MIL data points are grouped into bags and only the bag labels are available. Formally, the input data points $x_1, ..., x_n$ are grouped into some bags $B_1, ..., B_m$, each being associated with a bag label $Y_I = \{-1, 1\}, I \subseteq \{1, ..., m\}$[1]. The assumption of MIL is that there is at least one positive instance in a positive bag and all instances in negative bags are negative. In our case, in positive and negative bags are positive and negative patches respectively which contain sub-patches as instances. This assumption can be interpreted as two linear constraints for optimization.

$$\sum_{i \in I} \frac{y_i + 1}{2} \geq 1, \quad s.t. \quad Y_I = 1 \tag{6.22}$$

$$y_i = -1, \quad s.t. \quad Y_I = -1. \tag{6.23}$$

Based on these two assumptions, the objective is to identify the witness instance of each positive bag. The concept of margin maximization in SVM is extended from instance margin to bag margin, which results in a Multiple Instance (MI)-SVM. The margin of a bag with respect to a hyperplane is defined as

$$D_I = Y_I \max_{i \in I} (\boldsymbol{w}^T x + b) \tag{6.24}$$

In other words, the margin of a bag is the maximum distance between the hyperplane and all of its instances. Based on the notation of bag margin, MI-SVM is formulated as

$$\begin{aligned} \min_{\boldsymbol{w}, b, \boldsymbol{\xi}} \quad & \frac{1}{2} \|\boldsymbol{w}\|^2 + C \sum_{I=1}^{m} \xi_I \\ \text{s.t.} \quad & Y_I \max_{i \in I} (\boldsymbol{w}^T x_i + b) \geq 1 - \xi_I, \quad \xi_I \geq 0 \end{aligned} \tag{6.25}$$

[1] We consider only two-class classification problem and this can be easily extended to multi-class classification problem.

Data: The training bags $D = \{B_1, ..., B_m\}$ and bag labels $L = \{l_i, ..., l_m\}1$
Result: The bag-level classifier C-SVM
begin
  $P = \{X_I^0 = \frac{1}{|I|}\sum_{i\in I}^{x_i} |Y_I = 1\}$;
  $N = \{x_i | Y_I = -1, i \in I\}$;
  repeat
    $P^k = P$;
    $S = P^k \cup N$;
    Train a classifier C-SVM using samples S ;
    $P = \phi$;
    forall the *positive bags* B_I **do**
      $y_i = \text{sgn}\,(f(x_i)), \quad i \in I$;
      $X_I^k = \max_{x_i \in B_I} f(x_i)$;
      Add X_I^k to P ;
    end
  until $P^k = P$;
end

Algorithm 6: MI-SVM pseudocode

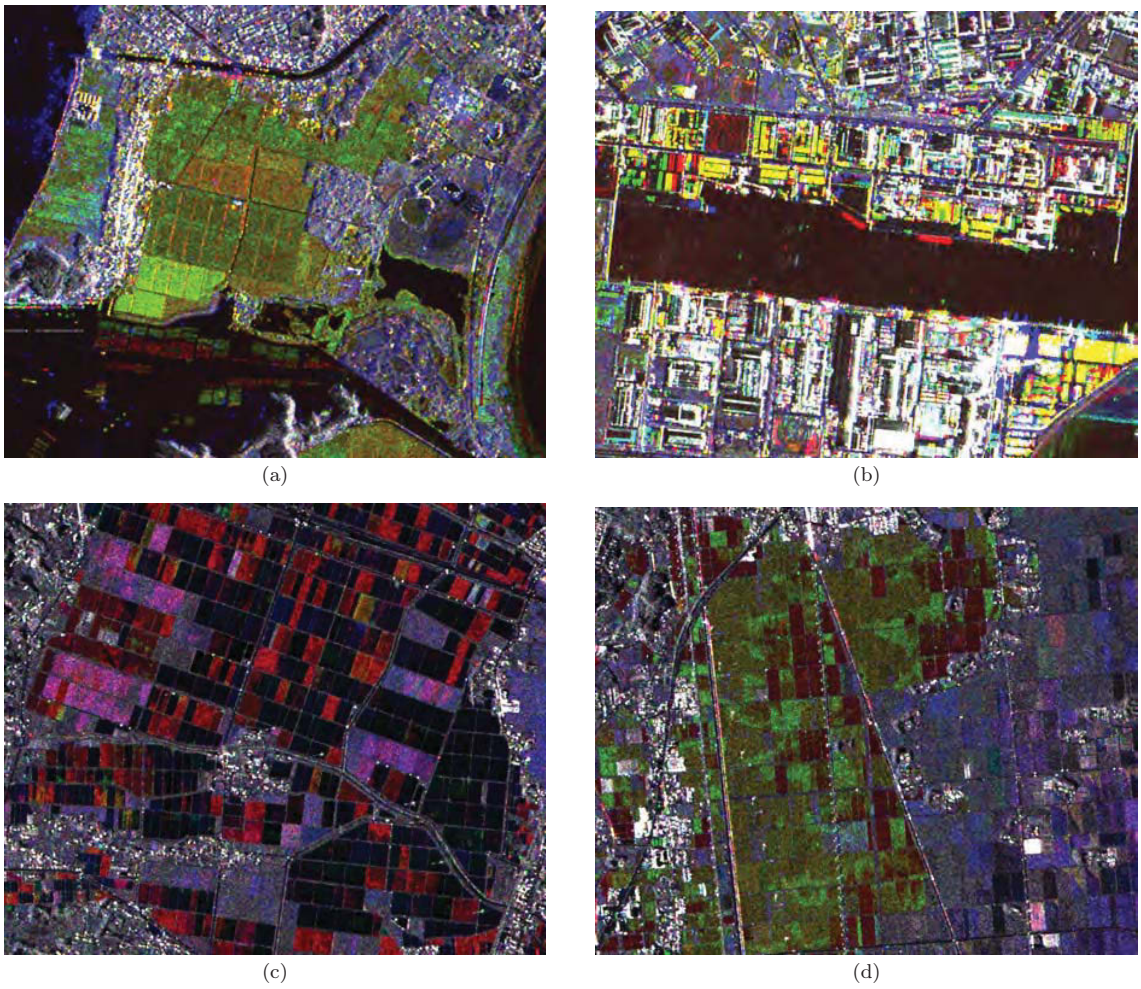
For a negative bag, the constraints can be decomposed for each instance as $-\boldsymbol{w}^T x_i - b \geq 1 - \xi_I$ with $i \in I$ and $Y_i = -1$. For a positive bag, a selector variable $S(I) \in I$ is defined to denote the index of the most positive ("witness") instance in the positive bag. All instances which are not selected will be discarded and have no influence on the learning. Therefore, the constraint for a positive bag can be rewritten as $\boldsymbol{w}^T x_{S(I)} + b \geq 1 - \xi_I$ with $Y_i = 1$.

Nevertheless, the optimization in Eq. (6.25) is a mixed integer programming problem, which cannot be solved efficiently by the methods published in the literature. To find a global optimization, we have to check all possible assignments of the labels to the instances in the positive bags and for each assignment, the bag margin has to be evaluated. For a medium size configuration, this combinatory problem becomes infeasible to solve. Therefore, we adopt the heuristic optimization proposed by Andrews *et al.* [2002]. Starting from the mean instances of the positive bags as positive instances, an alternating optimization between the update of the classifier and the identification of the witnesses is performed. In each iteration, the most positive elements in the positive bags are selected as witnesses. The algorithm is summarized in Alg. 6.

6.5 Visualization of SAR Image Time Series

In the case of multi-temporal SAR image information mining, an important topic is to develop efficient methods for image sequence visualization. Conventional methods involve dimension reduction, which is a well-established research topic and many methods are available. One of the biggest problems in applying these methods is that they distort the information, which means that an expert user cannot easily recognize the image content from the representation after dimension reduction. This makes it hard to use pseudo color display techniques. Furthermore, all these methods lose a certain amount of information, and not all information is visible to the human eye.

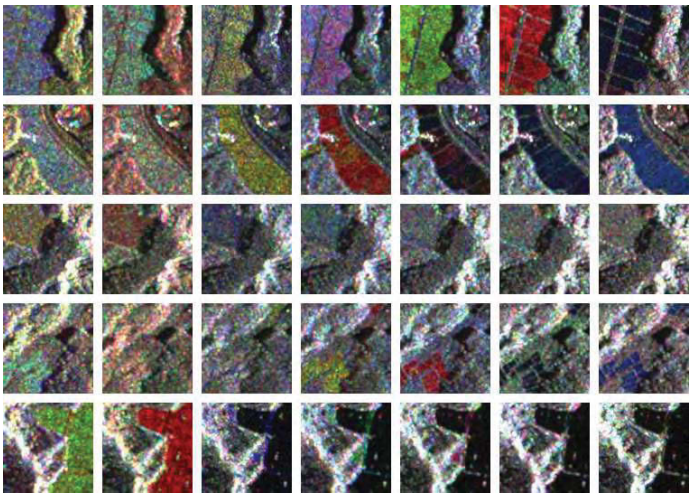
Therefore, we use a simple animation representation in our approach. However, if we concatenated on gray scale SAR images, an expert user's eyes would become exhausted after a few hours and the reliability of manual supervision would decrease significantly. In addition, an animation



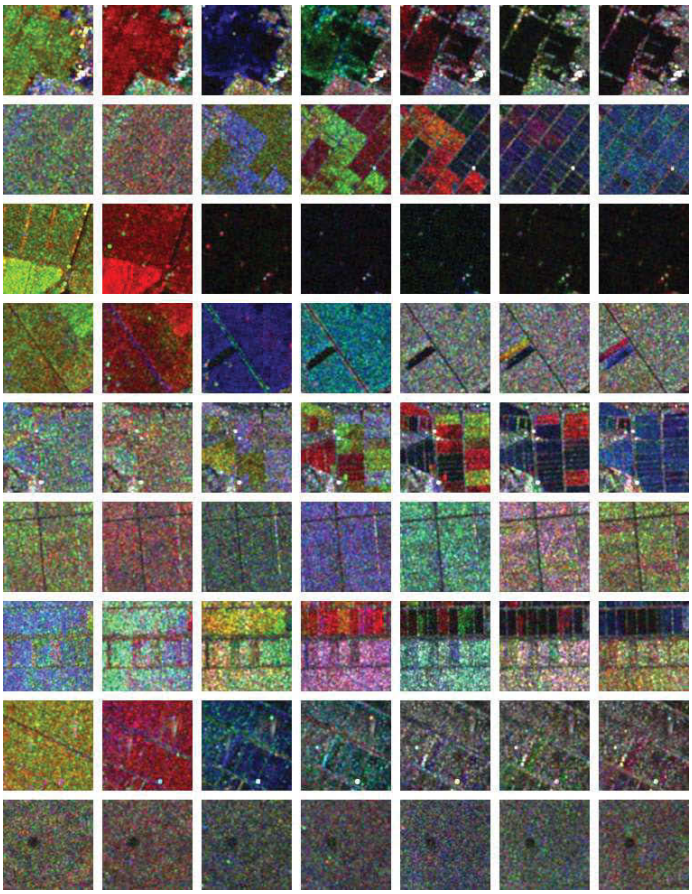
(a) (b) (c) (d)

Figure 6.10: Color representation of multi-temporal TerraSAR-X images covering Sendai, Japan.

composed of gray scale images cannot highlight the content variations in multi-temporal SAR images; however, the human eye is more sensitive to color variation than gray scale variation. Thus, to highlight the content variation, triples of successive SAR images are concatenated and represented as a color image, This method is applied to all the successive images in a given sequence. Formally, given a sequence of N images $\mathcal{I} = \{I_1, I_2, ..., I_N\}$, $N-2$ color images $\mathcal{I}^c = \{I_i^c \mid I_i^c = (I_i, I_{i+1}, I_{i+2}), i = 1, 2, .., N-2\}$ can be generated by concatenating triples of successive images. The advantage of this representation is that all the information remains visible and, most importantly, there is no information distortion. In addition, the content variation between images is highlighted, which makes a better interpretation feasible. Four examples of color representation of multi-temporal TerraSAR-X images are shown in Fig. 6.10. These images were acquired during the earthquake and tsunami disaster that occurred in Japan in 2011. Example patches of two classes, i.e. *mountains* and *agricultural fields*, are shown in Fig. 6.11. Compared with the gray scale visualization shown in Fig. 1.1, the color representation is much better suited for highlighting the temporal variations. With this simple but powerful color representation, four kinds of mountains and nine kinds of agricultural fields can be well discriminated



(a) Different mountains



(b) Different fields

Figure 6.11: Color representation of temporal patterns of mountains and agricultural fields.

visually. Obviously, our color representation can significantly highlight any content variations while not distorting the information, which greatly facilitates the image interpretation. Without

any processing, we can easily observe many temporal patterns and content variations become completely visible, which shows the powerful capability of visual data mining. As each color shown in the image represents a kind of temporal pattern, it can be seen from Fig. 6.10(a)(c)(d) that there are different temporal patterns due to flooding within a common area. In addition, Fig. 6.10(b) shows a part of Sendai harbor, where a lot of basic infrastructure was destroyed in the disaster.

6.6 Implementation

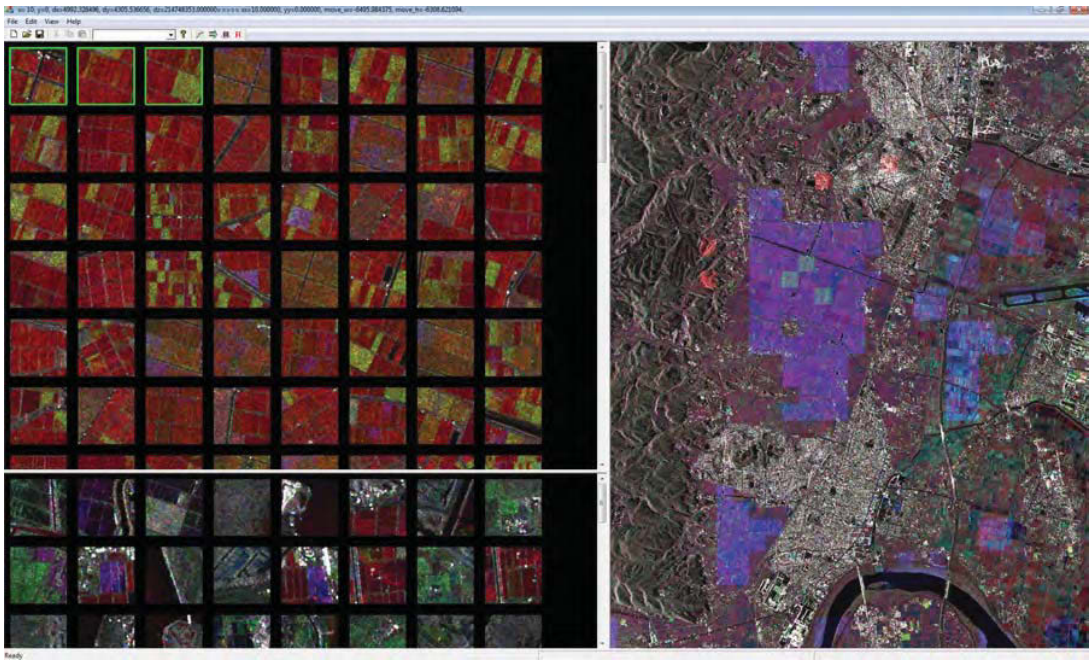
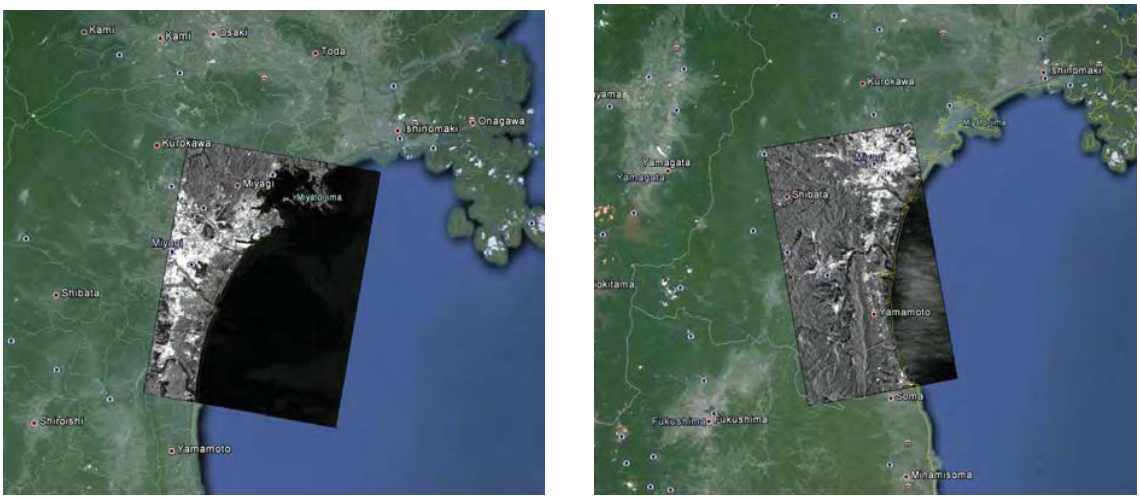


Figure 6.12: The interface of the cascaded active learning system.

The entire system mainly consists of three components, which are feature extraction, visualization, and cascaded active learning. The most important part is cascaded active learning, which lies at the heart of the entire system. It is composed of several levels and has a hierarchical structure. Each level is associated with a specific patch size. In the hierarchy, the patch size decreases from lower level to higher level. Spatial and temporal features are extracted at all levels, which can be organized using a quad-tree. The implemented system shown in Fig. 6.12 has three main panels. The big panel on the right side is responsible for the visualization of the entire scene because sometimes it is not easy to recognize a single patch without seeing its surrounding context. The upper panel displays the current positive patches in the current iteration while the lower left panel shows the patches that are close to the decision boundary, which are important for active learning. The individual images acquired at the same time that constitute the patches in these two panels are displayed at the same time such that an expert user can visually check the similarity of the temporal patterns. The system starts from the samples selected by an expert user at the first level. At each level, active learning is applied to speed up the convergence of the annotation. Active learning is performed until the annotation is satisfactory, which can be verified through visual checks as supervision. Then, the system moves

to the next level where samples are automatically selected through multiple instance learning. At this point, an expert user can manually edit the samples followed by active learning again. One of the important improvements of our system is to discard all negative patches and to classify only the patches that are offsprings (or sub-patches) of the patches that have been classified as positive at the previous level. At the end, one class of patches can be exported and used as reference data. This process is repeated until the last level has been reached. As we classify only the sub-patches, the speed can be improved significantly when we limit ourselves to the relevant patches. Another benefit of this coarse-to-fine strategy is the refinement of the classes, as shown for *water* in Fig.6.1, which is of practical significance in generating ground truth for semantic learning. It can also be used to discover and explore patterns in multi-temporal images.



(a) Ascending orbit (b) Descending orbit

Figure 6.13: Google Earth overlay of the disaster area.

6.7 Evaluation and Discussion

To evaluate our method, two real datasets covering Sendai, Japan, before and after the disaster in 2011 have been prepared. Using these two datasets, we have compared cascaded active learning with SVM active learning that operates only at the last level. Both datasets and evaluations are described in the following section.

6.7.1 Dataset and Setup

On march 11th, 2011, a devastating disaster was caused by a magnitude 9.03 on the moment magnitude (Mw) scale undersea earthquake off the coast of Japan. The earthquake triggered powerful tsunami waves that reached heights of up to 40.5 m in Miyako in the Tohoku Iwate Prefecture, and which, in the Sendai area, travelled up to 10 km inland.

The temporal intervals of these two datasets are around 11 days; their detailed parameters are listed in Tables 6.1 and 6.2. The images in each dataset have exactly the same resolution and number of looks in both range and azimuth direction. Also, the image incidence angles are quite close. The devastating tsunami destroyed a lot of constructions close to the Sendai airport and

resulted in various temporal patterns. Therefore, these two datasets are quite good candidates for SAR image time series. Typical subsets of these two data sets are shown in Fig. 6.10.

As the images of a given orbit branch have the same geometrical imaging parameters, it is very reasonable to assume that there is no rotation and only translation between them. For the purpose of co-registration, 10 corresponding strong point scatterers were selected from each image to determine their horizontal and vertical translation. The resulting co-registration error was less than one pixel. The two datasets were then tiled on three levels with patch sizes ranging from 200×200 to 50×50 pixels. If the patch size is too small, it is not easy to visually discriminate the patch content. BoSTW feature vectors were extracted from all three levels. The dictionary size was set to 200 elements, and the window size for local feature extraction was 3×3 pixels. We did not consider overlapping patches and the temporal window size was assumed to be the same as the number of images because the time span of the data set is very short, which is around 3 months (the temporal window size could be shorter for a long SAR time series). The reference data we used were generated using an active learning system under human supervision.

The accuracy measures we used for evaluation are *precision*, *recall*, and *F-score* together with *computational complexity* in terms of seconds for learning and classification. *Precision* is the fraction of retrieved images that are relevant to the search and *recall* is the fraction of the images that are relevant to the query and that are successfully retrieved. *F-score* that combines precision and recall is the harmonic mean of precision and recall, where recall and precision are evenly weighted. We selected SVM active learning performed only on the last level as a baseline for comparison with cascaded active learning.

$$\text{precision} = \frac{|\{\text{relevant images}\} \cap \{\text{retrieved images}\}|}{|\{\text{retrieved images}\}|} \tag{6.26}$$

$$\text{recall} = \frac{|\{\text{relevant images}\} \cap \{\text{retrieved images}\}|}{|\{\text{relevant images}\}|} \tag{6.27}$$

$$\text{F-score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{6.28}$$

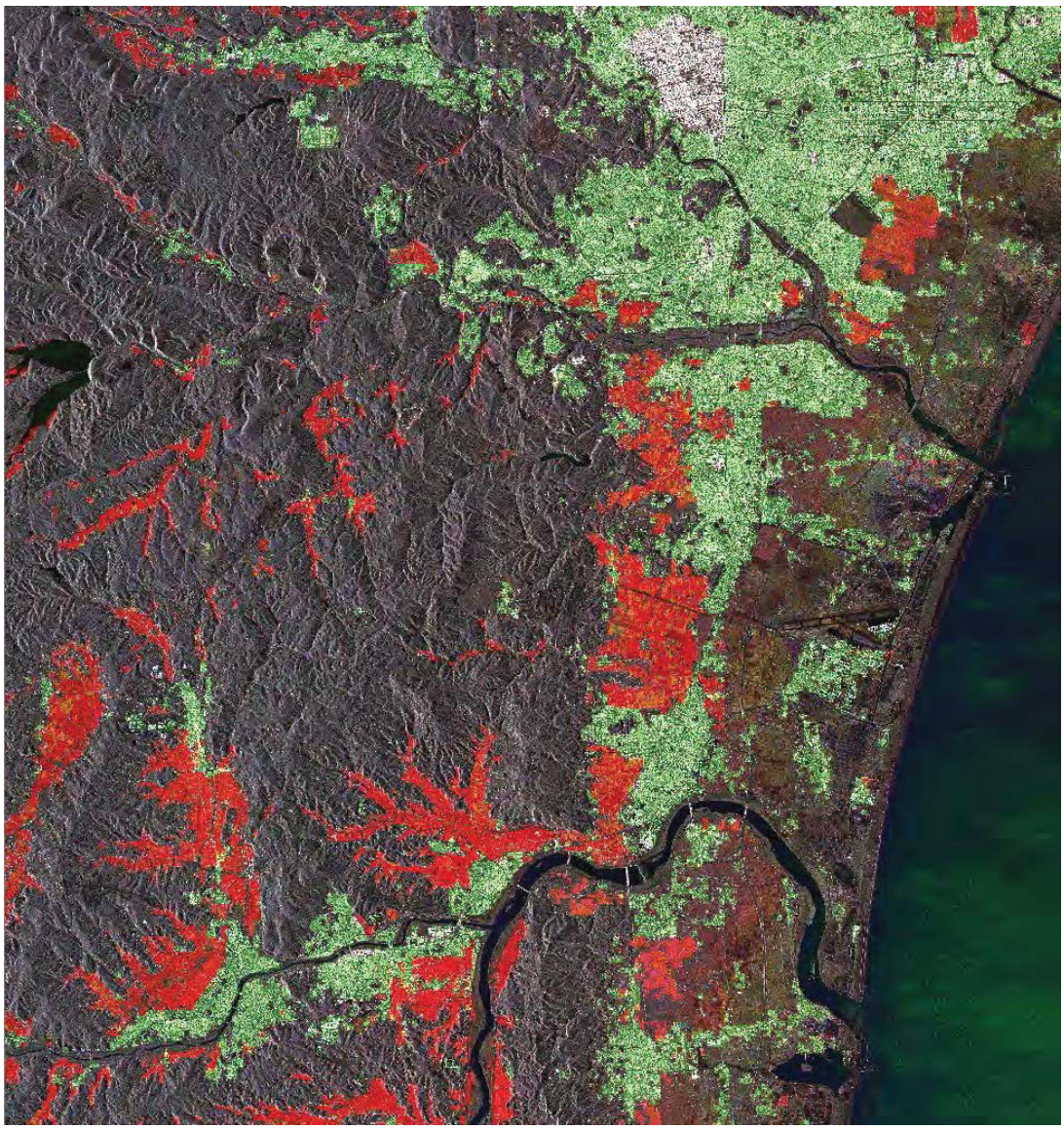


Figure 6.14: Annotation of *flooding* and *houses* in the ascending branch dataset.

Figure 6.15: Annotation of *flooded fields*, *agricultural fields*, *beaches*, and *mountains* in the descending branch dataset.

Table 6.1: Ascending branch data set.

Image	acc. time	Inc. angle	Height (km)	R. P. Spac.. (m)	A. P. Spac. (m)	R. looks	A. looks
dims_op_oc_dfd2_369486385_1	2009.10.08	35.27	195	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486405_1	2011.03.31	35.21	153	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486421_1	2011.04.11	35.21	162	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486461_1	2011.04.22	35.21	159	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486477_1	2011.05.14	35.22	161	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486493_1	2011.05.25	35.24	164	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486509_1	2011.06.05	35.22	157	5.88	5.90	2.88	2.04
dims_op_oc_dfd2_369486525_1	2011.06.16	35.25	159	5.88	5.90	2.88	2.04

Table 6.2: Descending branch data set.

Image	acc. time	Inc. angle	Height (km)	R. P. Spac.. (m)	A. P. Spac. (m)	R. looks	A. looks
dims_op_oc_dfd2_369487881_1	2008.09.21	37.32	55	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369487919_1	2010.10.20	37.32	55	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369487971_1	2011.03.12	37.30	54	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369488007_1	2011.03.23	37.32	55	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369488043_1	2011.05.06	37.30	54	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369488081_1	2011.05.17	37.30	54	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369488118_1	2011.05.28	37.30	54	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369488155_1	2011.06.08	37.30	54	5.78	5.75	2.98	1.99
dims_op_oc_dfd2_369488207_1	2011.06.19	37.30	54	5.78	5.75	2.98	1.99

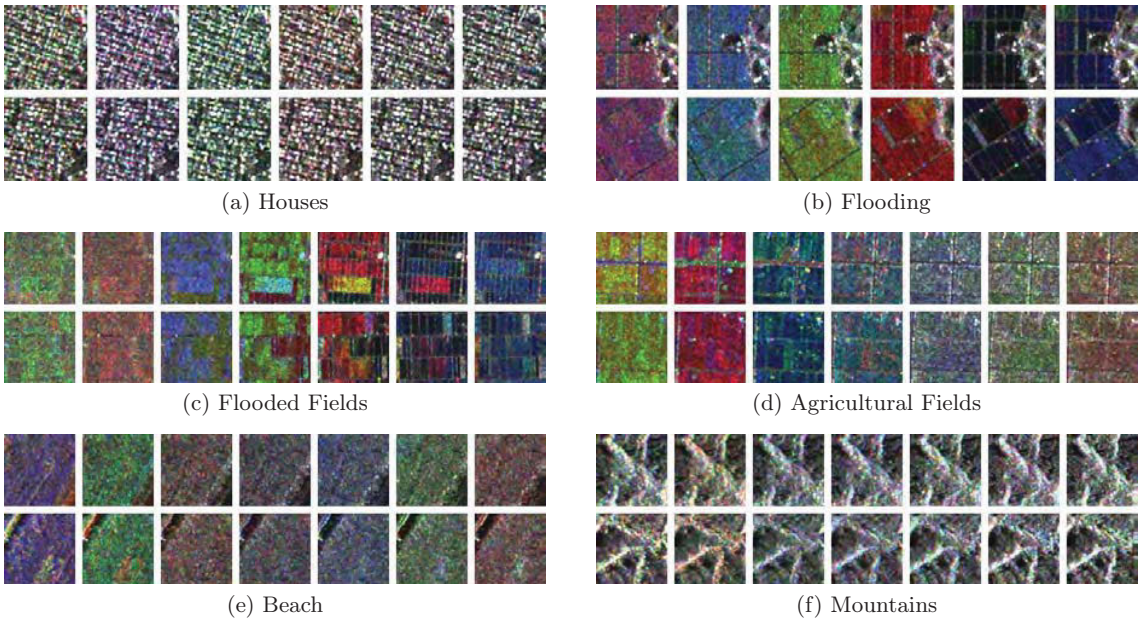
6.7.2 Experiments

We have evaluated the method and the system through SAR ITS temporal pattern retrieval. Six kinds of temporal patterns shown in Fig. 6.16 are selected for retrieval. Among the six classes, *flooding*, *flooded fields*, and *agricultural fields* are dynamic temporal patterns and the other three classes, namely *houses*, *beach*, and *mountain*, are stable classes. The first two classes are selected from the descending branch dataset and the remainder is taken from the ascending branch dataset. For each class, sample patches with a size of 200×200 pixels are selected manually to initialize the system. In total, 10 sample patches are selected in the beginning and 20 patches that are close to the decision boundary are selected and annotated in each iteration of active learning. 10 iterations are performed at each level, thus, totally 3×10 iterations are performed. In order to obtain fair comparisons, 30 iterations of the baseline SVM active learning method are also performed on the last level with a patch size of 50×50 pixels. In each iteration, the same number of 20 patches are manually labeled and added to the training data and, precision, recall, and F-score are computed. In addition, the time consumed by the training and classification is recorded. It is worth noting that there are two ways to compute the accuracy measures, depending on the reference data. One way is to compute the accuracy measures at each level separately. In this case, the reference data at levels below the last one are the patches that contain any true retrieved patches in the reference data of the last level. Another way is to compute the accuracy measures with respect to the true patches of the last level. In this case, the recall should always decrease. There should be no big difference in the accuracy of the last level between these two alternatives. In our evaluation, we adopted the first one for computing the accuracy.

All three accuracy measures for the six classes are shown in Fig. 6.17, Fig. 6.18, and Fig. 6.19 respectively. The corresponding computing times are shown in Fig. 6.20. The corresponding annotations of the six classes on the large scenes are shown in Fig. 6.14 and Fig. 6.15. The most important observation is that the computation burden of our cascaded active learning has been significantly reduced compared with the baseline, where the computation increases linearly with respect to iterations. This can be well explained by the motivation of cascaded active learning, which is to discard as many as possible irrelevant patches at lower levels and focus the computation on the relevant patches at higher levels. Although the number of training samples is the same, there are fewer patches to be classified compared with SVM active learning. Therefore, we can quickly find the temporal patterns in SAR ITS. For the first two classes shown in Fig. 6.20, there is a jump while moving from one to the next level. This is because each parent is tiled into four sub-patches at the next level. Compared with the previous level, there are 4 times more patches to be classified. As the number of patches for the first two classes is quite large, there is a jump in computational effort. For the remaining four classes, the speed jump is not apparent and the computational time does not increase with respect to the iterations.

On the other hand, it can be seen that all the three accuracy measures of cascaded active learning are generally higher than the baseline. As the initial training samples are randomly selected, there are probably some fluctuations in the first iterations. After that, the accuracy increases steadily until reaching a final value. There is also a jump in precision and recall. The reason is the same as for the jump in computational effort, as there are more patches to be classified as the level goes up. However, the accuracy at the last level is always better than our baseline method because SVM active learning is performed only at the last level with a patch size of 50×50 pixels. On the contrary, cascaded active learning discards all irrelevant patches at the first two levels; thus, a good classifier learns easily with the confusion of other classes, which happens to the baseline. As precision and recall depict only one aspect of the retrieval,

the F-score would be a better accuracy measure to demonstrate the overall accuracy. We can see from Fig. 6.19 that there is not much fluctuation in the F-score shown, and the cascaded active learning is always better than SVM active learning.

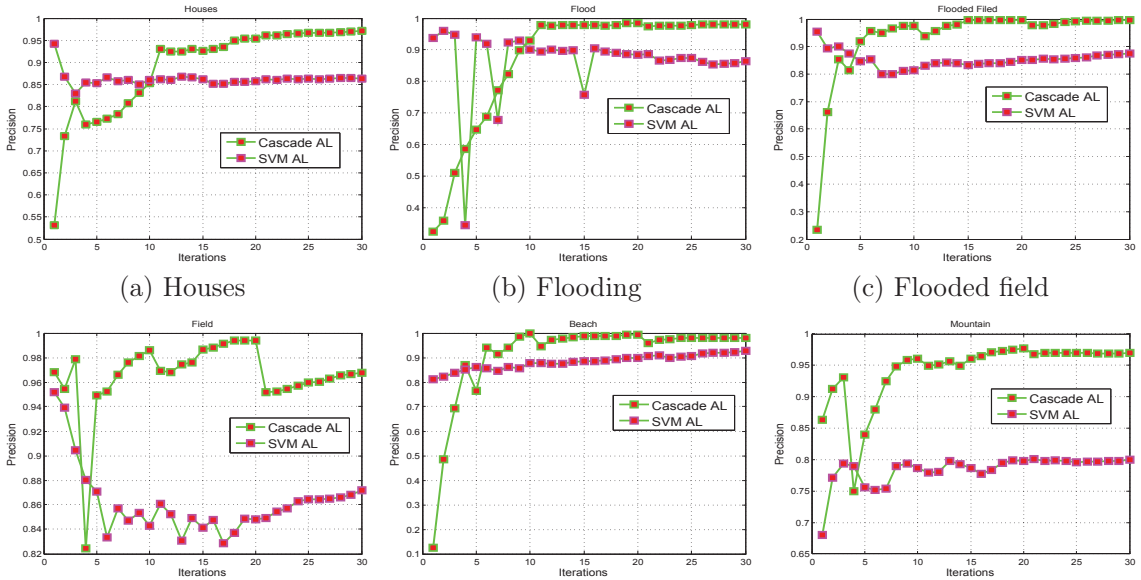


(a) Houses (b) Flooding

(c) Flooded Fields (d) Agricultural Fields

(e) Beach (f) Mountains

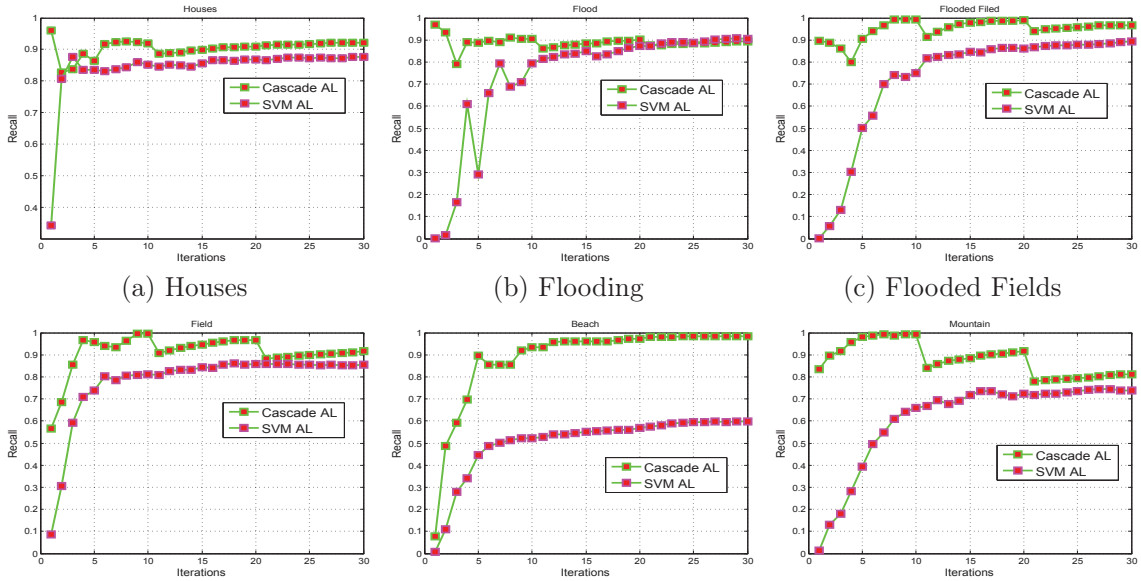
Figure 6.16: Color representation of example classes for retrieval.



(a) Houses (b) Flooding (c) Flooded field

(d) Agricultural Fields (e) Beach (f) Mountains

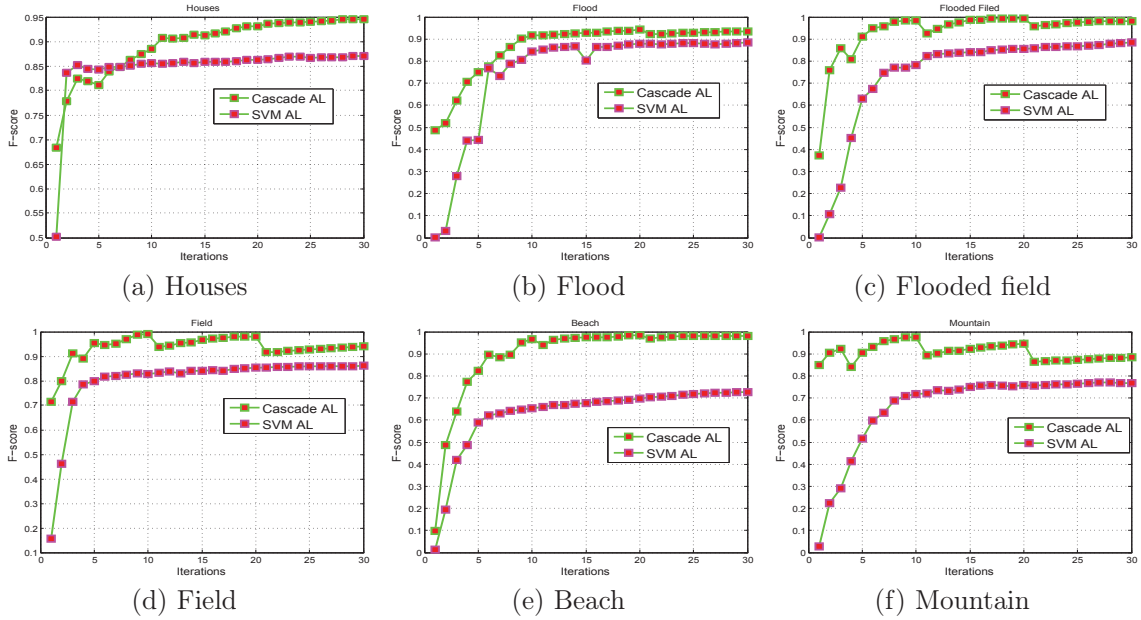
Figure 6.17: Precision of the six classes. It is obvious that cascaded active learning performs better in precision than SVM active learning. Despite the precision fluctuations in the first iterations, cascaded active learning converges quite fast after the first level.



(a) Houses (b) Flooding (c) Flooded Fields

(d) Agricultural Fields (e) Beach (f) Mountains

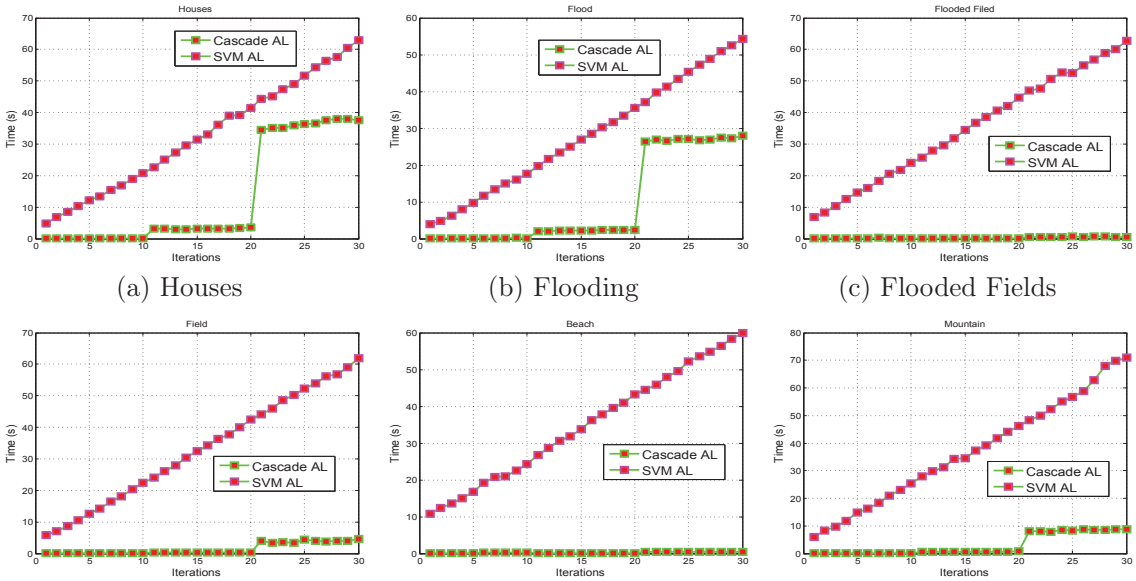
Figure 6.18: Recall of the six classes. Apparently, the recall of cascaded active learning is better or the same in the worst case. The jump in F-score occurs when moving to a new level because all the negative patches are discarded. If the discarded negative patches have some positive target, the recall would decrease when moving to a new level.



(a) Houses (b) Flood (c) Flooded field

(d) Field (e) Beach (f) Mountain

Figure 6.19: F-score of the six classes. It can be seen clearly that cascaded active learning always has a better F-score than SVM active learning. The reason for the jump in F-score is the same as for recall. As F-score is an overall performance measure, it does not have any fluctuations.



(a) Houses (b) Flooding (c) Flooded Fields

(d) Agricultural Fields (e) Beach (f) Mountains

Figure 6.20: Computing time for the six classes. Obviously, the computing time of SVM active learning increases linearly with respect to iterations. In contrast, the computing time of cascaded active learning does not increase as the level goes up for classes with a medium number of patches. For large classes, like *flooding* and *houses* of the ascending branch dataset, the computing time of cascaded active learning increases because there are many more patches that have to be classified when moving to a new level.

6.8 Summary

In this chapter, a cascaded active learning method relying on a coarse-to-fine strategy has been proposed for spatial and temporal SAR image information mining, which allows fast indexing and the discovery of hidden spatial and temporal patterns. Based on the observation that every class covers only a certain part in a large image, it is not really necessary to process all image patches because only some of them are relevant to the queried class. The learning method should discard all irrelevant patches as early as possible by a weak classifier that can be learned quickly and focus the training and learning on the relevant patches. In this way, the computational burden can be significantly decreased and the target class can be discovered quickly. With this aim, the cascaded active learning works on different levels. Each level is associated with a specific patch size and all levels are organized in a quad-tree structure. At each level, a relevance feedback classification based on SVM active learning is performed, in which the most informative samples to the classifier are selected and manually labeled. All irrelevant patches are discarded at the first few levels through active learning, which can then gradually discover any hidden temporal patterns. All negative patches are discarded and the classification at the next level is performed only on the positive patches that remain of the previous level; thereby, the data volume that needs to be processed decreases remarkably. Obviously, this method benefits significantly from the speed-up in learning. In addition, a simple yet efficient color animation for multi-temporal SAR image visualization has been proposed, which does not distort the image content while highlighting the content variation.

A multi-temporal SAR data set consisting of TerraSAR-X images acquired during the 2011 tsunami disaster in Japan has been prepared and used for evaluation. Three levels have been prepared and Bag-of-Spatial-Temporal-Words (BoSTW) features have been extracted from the multi-temporal SAR images. As a baseline for comparison, SVM active learning has been applied only on the last level. Three accuracy measures and the run time have been compiled for comparison. It has been demonstrated that cascaded active learning can only achieve better accuracy, but also reduces the computing time remarkably. The cascaded active learning method has been implemented in a practical system, which allows quick indexing of temporal patterns in multi-temporal SAR images.

Chapter 7

Conclusions

In this thesis, we propose some new methods for high resolution spatial and temporal SAR image information mining based on the intrinsic properties of SAR images. This work is mainly composed of three parts. The first part focuses on bi-temporal image analysis, which is a basic topic in the field of multi-temporal analysis. In this part, we apply information similarity measures to SAR change detection. The second part concentrates on feature extraction from high resolution SAR images for patch level classification. In the third part, a cascaded active learning approach is developed for temporal pattern mining in SAR ITS. A summary and discussion of these three parts are presented in this section.

In chapter 4, starting from statistical models, we present practical models and estimation methods for high resolution SAR images. We do not only need accurate models but also reliable estimation methods, either analytical or numerical ones. The Method of Log Cumulants (MoLC) is applied as an estimation method that performs better than conventional approaches, and which boils down to solving constrained equations. A Levenberg-Marquardt algorithm, which shows high reliability and robustness, is used to solve the constrained optimization problems. We perform a comprehensive evaluation of the statistical models by using a quite diverse dataset consisting of 3582 TerraSAR-X image patches selected from 20 target classes.

Based on the statistical models, we apply the information similarity measures to multi-temporal SAR change detections in the spatial and wavelet domain. To evaluate their performance, a benchmark dataset is created by simulating changes, such as statistical changes in first, second, and higher order statistics, which compensates the lack of a common benchmark dataset for evaluating various methods. A comprehensive evaluation of information similarity measures using a synthetic dataset and a real dataset is performed. Through this evaluation, we found that Kullback-Leibler divergence performs quite well in detecting changes in intensity. In contrast, mutual, variational, and mixed information are better alternatives for detecting changes in second order and higher order statistics. These information similarity measures can also be used as a feature vector in analyzing SAR ITS.

Due to the statistical similarity between SAR images and wavelet coefficients, our statistical models are applied to SAR change detection in the wavelet domain. Both GGD and GΓD are investigated for wavelet coefficient modeling because of their good mathematical properties. Different estimation methods are employed for the determination of actual parameters governing the probability density function. Our study concludes that the estimation accuracy depends very much on the estimation method although the applied model is important, too.

One problem in applying these similarity measures to change detection is that a closed form expression should be available. Otherwise, we have to resort to numerical methods, which would increase remarkably the computational burden, thereby decreasing the applicability of the meth-

ods. However, this depends heavily on the functional form of the probability density function. Thus, statistical models with good analytical properties that can be easily estimated are preferable in practical applications, such as the generalized Gamma distribution.

Through this study, we also found some remaining problems on this topic that should be further investigated. One is that the problem of change detection is not well formulated mathematically, although many solutions have been proposed. It would be good to formulate the problem before developing new methods. Without a good formulation, it is quite hard to describe various changes mathematically and to find efficient methods for change detection. Another advanced subject is the categorization of different changes, which is worth investigating further. However, a large data base of different kinds of changes is needed, although we have simulated several kinds of changes.

In chapter 5, based on the intrinsic characteristics of VHR SAR images, two new feature extraction methods are developed. The first one is a new feature extraction method for the structure description of high resolution SAR images, which is inspired by the ratio edge detector. Ratios in various directions in the local neighborhood are applied to enhance the Bag-of-Words (BoW) feature vector generated using only the local neighborhood statistics. As the ratios in the horizontal and vertical directions can be considered as an extension of image gradients, they are used to adapt a Weber Local Descriptor (WLD) to SAR images, which is a joint histogram of two components: differential excitation and orientation. The second method is a simple yet efficient feature extraction method within the Bag-of-Words (BoW) framework. It has two main innovations. Firstly and most interestingly, this method does not need any local feature extraction; instead, it uses directly the pixel values from a local window as low level features. Secondly, in contrast to many unsupervised feature learning methods, a random dictionary is applied to feature space quantization. We demonstrate that a random dictionary can play the same role in the BoW feature extraction as a dictionary learned by k-means clustering. The advantage of a random dictionary is that it does not lead to a significant loss of classification accuracy yet the time-consuming process of dictionary learning is avoided. In parallel, we developed a new feature coding method, called incremental coding. Altogether, the new feature extractor and the incremental coding can achieve significantly better SAR image classification accuracies than state-of-the-art feature extractors and feature coding methods.The BoW method has been extended to SAR Image Time Series (ITS) as well, resulting in a new Bag-of-Spatial-Temporal-Words (BoSTW) approach, which has demonstrated a better performance than a simple sequential concatenation of extracted texture features.

In addition, different aspects in the BoW model have been evaluated, such as patch size, patch sampling strategy, number of patches, universal or class-specific dictionaries, dictionary size, and feature coding methods. With smaller patch sizes, the image content variation can still be captured by the word histogram in the framework of BoW, because the word histogram counts the number of clusters that occur in a given image. Thus, a smaller patch size is still able to capture a large variation in image content by the word histogram. Regular dense sampling has shown higher accuracies than random sampling with the same number of patches. Increasing the number of patches in random sampling can improve the accuracy, but remains worse than regular sampling with small patches. A universal dictionary is better than a concatenation of class-specific dictionaries because the universal dictionary can capture the entire feature distribution rather than the individual feature spaces of each class. However, the computing time to generate a universal dictionary is much higher than that of class specific dictionaries. As for the dictionary size, there is no obvious gain in accuracy by increasing the dictionary size as long as it is sufficiently long. A large dictionary size would increase the computational burden. Thus, for a practical application, it would be better to choose an appropriate dictionary size in

terms of both accuracy and computing time. Although there are many methods trying to improve vector quantization by reducing the information loss, there is not much gain in accuracy. Our incremental feature coding method achieves significantly better results than state-of-the-art methods, even in the case of far less discriminative features. However, there are disadvantages as well. Incremental feature coding requires that the images should been labeled first. Hence, the sequence of classification and feature extraction has to be re-considered. Conventional pattern recognition first extracts features and then does a classification. However, this might not be the true mechanism of human perception, which is a complex process beyond current machine learning capabilities. From an opposite point of view, if we knew in advance the labels of the images, we could learn more discriminative features.

From this chapter, we conclude that the BoW method is very discriminative for SAR image classification. However, there may be still space for improvement because the BoW feature vectors are signal level image representations. Consequently, this representation is still far away from a semantic description. There is probably another level between the BoW representation and a semantic description because every local neighborhood used for local feature extraction has a fixed semantic meaning. To bridge this gap and to find the corresponding mapping between local neighborhoods and their semantics would be worth studying.

In chapter 6, we observed that every class covers only a small part of the image. Therefore, it is not necessary to process all patches because only some of them are relevant to the target class. The learning method should discard all irrelevant patches as early as possible and focus the training and learning on the relevant patches. In this way, not only can the accuracy be preserved, but also the computational burden can be significantly decreased. As a consequence, the target class can be discovered quickly.

Based on this principle, a cascaded active learning approach relying on a coarse-to-fine strategy for spatial and temporal SAR image information mining is developed, which allows fast indexing and the discovery of hidden spatial and temporal patterns in multi-temporal SAR images. A hierarchical image representation is adopted and each layer is associated with a specific patch size. The patches are tiled with smaller and smaller sizes. SVM active learning is applied on each level to reduce the manual effort for patch annotation. In this method, we have solved another problem, that is training sample propagation between levels, because the training samples selected manually by an expert user are only available at the first level. These samples cannot be used on all higher levels because the patches likely contain more than one class. Thus, they are not the desired class for all other levels. Two possible solutions exist. One is to ask an expert user to re-select manually training samples for a new level, which would be not acceptable to an expert user. The second solution is to automatically infer the samples for a new level from the manually selected samples at the first level, thereby further reducing the labor input. This would be preferable over the first solution, from an expert user's point of view. We apply multiple instance learning (MIL) to solve this problem, which considers each patch as a bag, with sub-patches at the next level as instances. The basic assumption of MIL is that there is at least one positive instance in each positive bag and all instances in a negative bag are negative. Thus, the difficulty in MIL stems from the label ambiguity, which leads to an integer programming problem. A heuristic method is used to solve this integer programming problem.

In addition, we developed a simple yet efficient visualization method for SAR ITS, which is simply a color animation that combines triples of successive images into color images. The advantage of this visualization compared with other dimension reduction methods is that it does not distort the image content such that all the information remains visible to the users. To evaluate this method, two TerraSAR-X images covering Sendai, Japan, before and after the tsunami in 2011, were selected. These two datasets are tiled into three levels with patch sizes

ranging from 200 to 50 pixels. Using an extension of the BoW method, called the BoSTW method, leads to feature extraction from all the three levels. Under this experimental setup, a cascaded active learning method is compared with a baseline SVM active learning method in terms of accuracy and computing time. The latter method operates only on the last level. Six classes of evolution patterns are selected for evaluation by retrieval. Three accuracy measures, i.e., precision, recall, and F-score, are employed for comparison as well as the computing time. From the experimental results, one can clearly see that the accuracy of cascaded active learning is always better (or the same in the worst case) than SVM active learning. The most important improvement is that the computational burden has been significantly reduced. The computing time of SVM active learning increases linearly with respect to the iterations. Interestingly, there is only a negligible increase in computing time for cascaded active learning for a medium class. For large classes, there is an obvious increase, but still much less than for the baseline. Between levels, there may be a jump in accuracy because of a new level. After moving to a new level, there may be more patches to be classified. Thus, the accuracy measures might have a jump.

Through the development of this method, we demonstrated the efficiency of cascaded active learning. However, there are still some drawbacks of this method. For example, if some patches have been discarded at the previous level, any of their sub-patches would not be considered any more at the next level, which would decrease the accuracy. Another drawback is that a multi-scale strategy is used to implement the coarse-to-fine method, which leads to inflexible borders between classes and prevents the users from finding and annotating evolution patterns with arbitrary shapes. To solve this problem for remote sensing images would be a great challenge.

Nomenclature

Roman Symbols

AUC Area Under Curve

BIC Bayesian Information Criterion

BoSTW Bag-of-Spatial-Temporal-Words

BoW Bag-of-Words

CDF Cumulative Distribution Function

CENTRIST Census Transform Histogram

CGF Cumulant Generating Function

ELBG Enhanced Linde-Buzo-Gray

EM Expectation-Maximization

EO Earth Observation

FAR False Alarm Rate

GΓD Generalized Gamma Distribution

GΓR Generalized Gamma Rayleigh

GGD Generalized Gaussian Distribution

GGR Generalized Gaussian Rayleigh

GLCM Gray Level Co-occurrence Matrix

GMM Gaussian Mixture Model

GMRF Gaussian Markov Random Field

ITS Image Time Series

KDE Kernel Density Estimation

KS Kolmogorov-Smirnov

LBP Local Binary Pattern

LDA Latent Dirichlet Allocation

LLC Locality-constrained Linear Coding

MGD Multilook Ground Detected

MGF Moment Generating Function

MIL Multiple Instance Learning

MLE Maximum Likelihood Estimation

MLPH Multilevel Local Pattern Histogram

MMSE Minimum Mean Squared Error

MoLC Method of Log Cumulant

MoM Method of Moment

PDF Probability Density Function

QMF Quadrature Mirror Filter

RCS Radar Cross Section

RIFT Rotation Invariant Feature Transform

ROC Receiver Operating Characteristic

SAR Synthetic Aperture Radar

SIFT Scale Invariant Feature Transform

STFT Short Time Fourier Transform

SURF Speeded Up Robust Feature

SVM Support Vector Machine

TPR True Positive Rate

UWT Undecimated Wavelet Transform

VHR Very High Resolution

WLD Weber Local Descriptor

Derivation of the CDF of a $\mathcal{K}$ distribution

In this section, we derive the closed form expression of Kullback-Leibler divergence between two generalized Gamma distribution

$$
\begin{aligned}
P(x) &= \frac{2\lambda L}{\Gamma(L)\Gamma(\alpha)}\Big(\lambda L x\Big)^{\frac{L+\alpha}{2}-1} K_{\alpha-L}(2\sqrt{\lambda L x}) \\
F(t) &= \int_0^t P(x)dx \\
&= \int_0^t \frac{2\lambda L}{\Gamma(L)\Gamma(\alpha)}\Big(\lambda L x\Big)^{\frac{L+\alpha}{2}-1} K_{\alpha-L}(2\sqrt{\lambda L x})dx \\
&= \frac{2\lambda L}{\Gamma(L)\Gamma(\alpha)}(\lambda L)^{\frac{L+\alpha}{2}-1}\int_0^t x^{\frac{L+\alpha}{2}-1} K_{\alpha-L}(2\sqrt{\lambda L x})dx \quad x = zt \\
&= \frac{2\lambda L}{\Gamma(L)\Gamma(\alpha)}(\lambda L t)^{\frac{L+\alpha}{2}-1} t\int_0^1 z^{\frac{L+\alpha}{2}-1} K_{\alpha-L}(2\sqrt{\lambda L z t})dz \\
&= \frac{2\lambda L}{\Gamma(L)\Gamma(\alpha)}(\lambda L t)^{\frac{L+\alpha}{2}-1} t\int_0^1 z^{\frac{L+\alpha}{2}-1} K_{\alpha-L}(2\sqrt{\lambda L t}\sqrt{z})dz
\end{aligned}
\tag{1}
$$

Based on the equation Eq.(2),

$$
\begin{aligned}
&\int_0^1 t^{\lambda}(1-t)^{\mu-1}K_{\nu}(a\sqrt{t})dt = \\
&2^{\nu-1}a^{-\nu}\frac{\Gamma(\nu)\Gamma(\mu)\Gamma(\lambda+1-\frac{\nu}{2})}{\Gamma(\lambda+1+\mu-\frac{\nu}{2})} \times {}_1F_2(\lambda+1-\frac{\nu}{2}; 1-\nu, \lambda+1+\mu-\frac{\nu}{2}; \frac{a^2}{4}) \\
&+ 2^{-\nu-1}a^{\nu}\frac{\Gamma(-\nu)\Gamma(\mu)\Gamma(\lambda+1+\frac{\nu}{2})}{\Gamma(\lambda+1+\mu+\frac{\nu}{2})} \times {}_1F_2(\lambda+1+\frac{\nu}{2}; 1+\nu, \lambda+1+\mu+\frac{\nu}{2}; \frac{a^2}{4})
\end{aligned}
\tag{2}
$$

we set $\lambda = \frac{L+\alpha}{2} - 1, \mu = 1, a = 2\sqrt{\lambda L t}, \nu = \alpha - L$, we get the equation Eq.(3)

$$\begin{aligned}\int_0^1 t^{\lambda} K_{\nu}(a\sqrt{t})dt = &\frac{1}{2}(\sqrt{\lambda L t})^{L-\alpha}\frac{\Gamma(\alpha - L)}{L} \times {}_1F_2(L; 1 + L - \alpha, L + 1; \lambda L t) \\ &+ \frac{1}{2}(\sqrt{\lambda L t})^{\alpha-L}\frac{\Gamma(L - \alpha)}{\alpha} \times {}_1F_2(\alpha; 1 + \alpha - 1, \alpha + 1; \lambda L t)\end{aligned} \tag{3}$$

Substituting Eq.(3) into Eq.(1)), we can derive the exact CDF of $\mathcal{G}^0$ distribution as shown in Eq.(4.17).

References

Abe, S. (2005). *Support Vector Machines for Pattern Classification (Advances in Pattern Recognition)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA. 116

Achim, A., Kuruoglu, E.E. & Zerubia, J. (2006). SAR image filtering based on the heavy-tailed Rayleigh model. *IEEE Trans. Image Process.*, **15**, 2686–2693. 13

Akbarizadeh, G. (2012). A New Statistical-Based Kurtosis Wavelet Energy Feature for Texture Recognition of SAR Images. *IEEE Trans. Geosci. Remote Sens.*, **50**, 4358–4368. 17

Alberga, V. (2009). Similarity Measures of Remotely Sensed Multi-sensor Images for Change Detection Applications. *Remote Sensing*, **1**, 122–143. 16

Andrews, S., Tsochantaridis, I. & Hofmann, T. (2002). Support Vector Machines for Multiple-Instance Learning. In *Proc. Advances in Neural Information Processing Systems NIPS*, 561–568. 124

Arsenault, H.H. & April, G. (1976). Properties of speckle integrated with a finite aperture and logarithmically transformed. *J. Opt. Soc. Am.*, **66**, 1160–1163. 13

Atto, A.M., Trouvé, E., E., Berthoumieu, Y. & Mercier, G. (2013). Multidate Divergence Matrices for the Analysis of SAR Image Time Series. *IEEE Trans. Geosci. Remote Sens.*, **51**, 1922–1938. 16

Bachmann, C.M., Ainsworth, T.L. & Fusina, R.A. (2006). Improved Manifold Coordinate Representations of Large-Scale Hyperspectral Scenes. *IEEE Trans. Geosci. Remote Sens.*, **44**, 2786–2803. 23

Bajcsy, P. & Groves, P. (2004). Methodology for Hyperspectral Band Selection. *Photogrammetric Engineering and Remote Sensing*, **70**, 793–802. 23

Ban, Y. & Yousif, O.A. (2012). Multitemporal Spaceborne SAR Data for Urban Change Detection in China. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 1087–1094. 13

Bastarrika, A., Chuvieco, E. & Martin, M.P. (2011). Automatic Burned Land Mapping From MODIS Time Series Images: Assessment in Mediterranean Ecosystems. *IEEE Trans. Geosci. Remote Sens.*, **49**, 3401–3413. 21

Bay, H., Ess, A., Tuytelaars, T. & Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.*, **110**, 346–359. 19

Bazi, Y., Bruzzone, L. & Melgani, F. (2005). An unsupervised approach based on the generalized Gaussian model to automatic change detection in multitemporal SAR images. *IEEE Trans. Geosci. Remote Sens.*, **43**, 874–887. 15

Bazi, Y., Bruzzone, L. & Melgani, F. (2007). Image thresholding based on the EM algorithm and the generalized Gaussian distribution. *Pattern Recognition*, **40**, 619 – 634. 15, 17

Bertsekas, D. (1976). On the Goldstein-Levitin-Polyak gradient projection method. *IEEE Trans. Autom. Control*, **21**, 174–184. 35

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA. 25

Blanchart, P., Ferecatu, M. & Datcu, M. (2011). Cascaded active learning for object retrieval using multiscale coarse to fine analysis. In *Proc. 18th IEEE International Conference on Image Processing (ICIP)*, 2793–2796, Brussels, Belgium. 23

Bombrun, L., Vasile, G., Gay, M. & Totir, F. (2011). Hierarchical Segmentation of Polarimetric SAR Images Using Heterogeneous Clutter Models. *IEEE Trans. Geosci. Remote Sens.*, **49**, 726–737. 13

Bovolo, F. & Bruzzone, L. (2005). A detail-preserving scale-driven approach to change detection in multitemporal SAR images. *IEEE Trans. Geosci. Remote Sens.*, **43**, 2963–2972. 15, 16

Bovolo, F. & Bruzzone, L. (2008). An Adaptive Technique based on Similarity Measures for Change Detection in Very High Resolution SAR Images. In *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, vol. 3, III–158 – III–161, Boston, Massachusetts, USA. 16

Bovolo, F., Bruzzone, L. & Marconcini, M. (2008). A Novel Approach to Unsupervised Change Detection Based on a Semisupervised SVM and a Similarity Measure. *IEEE Trans. Geosci. Remote Sens.*, **46**, 2070–2082. 15

Bovolo, F., Camps-Valls, G. & Bruzzone, L. (2010). A support vector domain method for change detection in multitemporal images. *Pattern Recogn. Lett.*, **31**, 1148–1154. 15

Bratasanu, D., Nedelcu, I. & Datcu, M. (2012). Interactive Spectral Band Discovery for Exploratory Visual Analysis of Satellite Images. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 207 –224. 23

Brooks, E.B., Thomas, V.A., Wynne, R.H. & Coulston, J.W. (2012). Fitting the Multitemporal Curve: A Fourier Series Approach to the Missing Data Problem in Remote Sensing Analysis. *IEEE Trans. Geosci. Remote Sens.*, **50**, 3340–3353. 21

Bruce, N.D. & Tsotsos, J.K. (2009). Saliency, attention, and visual search: an information theoretic approach. *Journal of vision*, **9**, 1–24. 82

Bruzzone, L. & Prieto, D.F. (2000). Automatic analysis of the difference image for unsupervised change detection. *IEEE Trans. Geosci. Remote Sens.*, **38**, 1171–1182. 15, 17

Bruzzone, L. & Prieto, D.F. (2001). Unsupervised retraining of a maximum likelihood classifier for the analysis of multitemporal remote sensing images. *IEEE Trans. Geosci. Remote Sens.*, **39**, 456–460. 15

Bruzzone, L. & Prieto, D.F. (2002). An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images. *IEEE Trans. Image Process.*, **11**, 452–466. 15, 16

Bruzzone, L. & Serpico, S.B. (1997). An iterative technique for the detection of land-cover transitions in multitemporal remote-sensing images. *IEEE Trans. Geosci. Remote Sens.*, **35**, 858–867. 15

Çelik, T. (2009). Multiscale Change Detection in Multitemporal Satellite Images. *IEEE Geosci. Remote Sens. Lett.*, **6**, 820–824. 17, 66

Çelik, T. (2010). A Bayesian approach to unsupervised multiscale change detection in synthetic aperture radar images. *Signal Process.*, **90**, 1471–1485. 15, 16, 17

Çelik, T. (2011). Bayesian change detection based on spatial sampling and Gaussian mixture model. *Pattern Recognition Letters*, **32**, 1635–1642. 17

Çelik, T. & Ma, K.K. (2010). Unsupervised Change Detection for Satellite Images Using Dual-Tree Complex Wavelet Transform. *IEEE Trans. Geosci. Remote Sens.*, **48**, 1199–1210. 17

Çelik, T. & Ma, K.K. (2011). Multitemporal Image Change Detection Using Undecimated Discrete Wavelet Transform and Active Contours. *IEEE Trans. Geosci. Remote Sens.*, **49**, 706–716. 17

Chatelain, F., Tourneret, J.Y., Inglada, J. & Ferrari, A. (2007). Bivariate Gamma Distributions for Image Registration and Change Detection. *IEEE Trans. Image Process.*, **16**, 1796–1806. 16

Chatfield, K., Lemtexpitsky, V., Vedaldi, A. & Zisserman, A. (2011). The devil is in the details: an evaluation of recent feature encoding methods. In *British Machine Vision Conference BMVC*. 86

Chen, H.M., Varshney, P.K. & Arora, M.K. (2003). Performance of mutual information similarity measure for registration of multitemporal remote sensing images. *IEEE Trans. Geosci. Remote Sens.*, **41**, 2445–2454. 16

Chen, J., Shan, S., He, C., Zhao, G., Pietikainen, M., Chen, X. & Gao, W. (2010). WLD: A robust local image descriptor. *IEEE Trans. Pattern Anal. Mach. Intell.*, **32**, 1705–1720. 74, 77

Chen, K.S., Wang, H.W., Wang, C.T. & Chang, W.Y. (2011). A Study of Decadal Coastal Changes on Western Taiwan Using a Time Series of ERS Satellite SAR Images. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 826–835. 15, 21

Choy, S.K. & Tong, C.S. (2010). Statistical Wavelet Subband Characterization Based on Generalized Gamma Density and Its Application in Texture Retrieval. *IEEE Trans. Image Process.*, **19**, 281–289. 63

Clausi, D.A. (2002). An analysis of co-occurrence texture statistics as a function of grey level quantization. *Canadian Journal of Remote Sensing*, **28**, 45–62. 102

Clausi, D.A. & Yue, B. (2004). Comparing cooccurrence probabilities and Markov random fields for texture analysis of SAR sea ice imagery. *IEEE Trans. Geosci. Remote Sens.*, **42**, 215–228. 17, 73

Coates, A. & Ng, A.Y. (2011). The Importance of Encoding Versus Training with Sparse Coding and Vector Quantization. In *Proc. 28th International Conference on Machine Learning ICML*, 921–928, Bellevue, WA. 89

Coates, A. & Ng, A.Y. (2012). Learning Feature Representations with K-means. In *Neural Networks: Tricks of the Trade*, vol. 7700 of *Lecture Notes in Computer Science*, 561–580, Springer Berlin Heidelberg. 90

Cross, G. & Jain, A. (1983). Markov Random Field Texture Models. *IEEE Trans. Pattern Anal. Mach. Intell.*, **5**, 25–39. 17

Cui, M., Razdan, A., Hu, J. & Wonka, P. (2009). Interactive Hyperspectral Image Visualization Using Convex Optimization. *IEEE Trans. Geosci. Remote Sens.*, **47**, 1673–1684. 23

Cui, S. & Datcu, M. (2012). Statistical Wavelet Subband Modeling for Multi-Temporal SAR Change Detection. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 1095–1109. 23

Cui, S., Dumitru, C.O. & Datcu, M. (2013a). Ratio-Detector-Based Feature Extraction for Very High Resolution SAR Image Patch Indexing. *IEEE Geosci. Remote Sens. Lett.*, **10**, 1175–1179. 73

Cui, S., Dumitru, C.O. & Datcu, M. (2013b). Semantic annotation in Earth observation based on active learning. *International Journal of Image and Data Fusion*, 1–23. 78, 90, 106

Dai, D., Yang, W. & Sun, H. (2011). Multilevel Local Pattern Histogram for SAR Image Classification. *IEEE Geosci. Remote Sens. Lett.*, **8**, 225–229. 74, 80

Daniela Espinoza-Molina, M.D., Dusan Gleich (2012). Evaluation of Bayesian Despeckling and Texture Extraction Methods Based on Gauss–Markov and Auto-Binomial Gibbs Random Fields: Application to TerraSAR-X Data. *IEEE Trans. Geosci. Remote Sens.*, **50**, 2001–2025. 13, 17

Datcu, M. & Seidel, K. (2005). Human-centered concepts for exploration and understanding of Earth observation images. *IEEE Trans. Geosci. Remote Sens.*, **43**, 601–609. 22

Delignon, Y. & Pieczynski, W. (2002). Modeling non-Rayleigh speckle distribution in SAR images. *IEEE Trans. Geosci. Remote Sens.*, **40**, 1430–1435. 14

Dempster, A., Laird, N. & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society, Series B*, **39**, 1–38. 36

Dietterich, T.G., Lathrop, R.H. & Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, **89**, 31 – 71. 112

Du, Q., Raksuntorn, N., Cai, S. & Moorhead, R.J. (2008). Color Display for Hyperspectral Imagery. *IEEE Trans. Geosci. Remote Sens.*, **46**, 1858–1866. 23

Dumitru, C.O., Singh, J. & Datcu, M. (2012). Selection of relevant features and TerraSAR-X products for classification of high resolution SAR images. In *Proc. 9th European Conference on Synthetic Aperture Radar, EUSAR.*, 243–246, Nuernberg, Germany. 80

Efros, A.A. & Freeman, W.T. (2001). Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, 341–346, ACM, New York, NY, USA. 49

Engdahl, M.E. & Hyyppa, J.M. (2003). Land-cover classification using multitemporal ERS-1/2 InSAR data. *IEEE Trans. Geosci. Remote Sens.*, **41**, 1620–1628. 20

Fallourd, R., Harant, O., Trouve, E., Nicolas, J.M., Gay, M., Walpersdorf, A., Mugnier, J.L., Serafini, J., Rosu, D., Bombrun, L., Vasile, G., Cotte, N., Vernier, F., Tupin, F., Moreau, L. & Bolon, P. (2011). Monitoring Temperate Glacier Displacement by Multi-Temporal TerraSAR-X Images and Continuous GPS Measurements. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 372–386. 21

Faur, D., Gavat, I. & Datcu, M. (2006). Mutual Information Based Measure for Image Content Characterization. In *Proceedings of the 11th Spanish Association Conference on Current Topics in Artificial Intelligence*, vol. 4177 of *CAEPIA'05*, 342–349, Springer-Verlag, Berlin, Heidelberg. 16

Feng, J., Cao, Z. & Pi, Y. (2013). Multiphase SAR Image Segmentation With G^0-Statistical-Model-Based Active Contours. *IEEE Trans. Geosci. Remote Sens.*, **51**, 4190–4199. 13

Fernando, B., Fromont, E., Muselet, D. & Sebban, M. (2012). Supervised learning of Gaussian mixture models for visual vocabulary generation. *Pattern Recogn.*, **45**, 897–907. 87

Figueiredo, M.A.F. & Jain, A.K. (2002). Unsupervised learning of finite mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.*, **24**, 381–396. 37

Frery, A.C., Muller, H.J., Yanasse, C.C.F. & Sant'Anna, S.J.S. (1997). A model for extremely heterogeneous clutter. *IEEE Trans. Geosci. Remote Sens.*, **35**, 648–659. 14, 30, 46, 56

Fukuda, S. & Hirosawa, H. (1999). A wavelet-based texture feature set applied to classification of multifrequency polarimetric SAR images. *IEEE Trans. Geosci. Remote Sens.*, **37**, 2282–2286. 17

Galland, F., Nicolas, J.M., Sportouche, H., Roche, M., Tupin, F. & Refregier, P. (2009). Unsupervised Synthetic Aperture Radar Image Segmentation Using Fisher Distributions. *IEEE Trans. Geosci. Remote Sens.*, **47**, 2966–2972. 13, 34

Gao, G. (2010). Statistical Modeling of SAR Images: A Survey. *Sensors*, **10**, 775–795. 13

Gao, G. & Shi, G. (2012). The CFAR Detection of Ground Moving Targets Based on a Joint Metric of SAR Interferogram's Magnitude and Phase. *IEEE Trans. Geosci. Remote Sens.*, **50**, 3618–3624. 13

Gao, G., Liu, L., Zhao, L., Shi, G. & Kuang, G. (2009). An Adaptive and Fast CFAR Algorithm Based on Automatic Censoring for Target Detection in High-Resolution SAR Images. *IEEE Trans. Geosci. Remote Sens.*, **47**, 1685–1697. 13

Gleich, D. & Datcu, M. (2007). Wavelet-Based Despeckling of SAR Images Using Gauss-Markov Random Fields. *IEEE Trans. Geosci. Remote Sens.*, **45**, 4127–4143. 17

Gleich, D. & Datcu, M. (2009). Wavelet-Based SAR Image Despeckling and Information Extraction, Using Particle Filter. *IEEE Trans. Image Process.*, **18**, 2167–2184. 17

Goldberger, J., Gordon, S. & Greenspan, H. (2003). An efficient image similarity measure based on approximations of KL-divergence between two gaussian mixtures. In *Proc. Ninth IEEE International Conference on Computer Vision, ICCV*, 487–493, Nice, France. 44, 50

Grobler, T.L., Ackermann, E.R., Olivier, J.C., van Zyl, A.J. & Kleynhans, W. (2012). Land-Cover Separability Analysis of MODIS Time-Series Data Using a Combined Simple Harmonic Oscillator and a Mean Reverting Stochastic Process. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 857–866. 21

Grobler, T.L., Ackermann, E.R., van Zyl, A.J., Olivier, J.C., Kleynhans, W. & Salmon, B.P. (2013). Using Page's Cumulative Sum Test on MODIS Time Series to Detect Land-Cover Changes. *IEEE Geosci. Remote Sens. Lett.*, **10**, 332–336. 21

Gu, M. & Abraham, D.A. (2001). Using McDaniel's model to represent non-Rayleigh reverberation. *IEEE J. Ocean. Eng.*, **26**, 348–357. 30

Gueguen, L. & Datcu, M. (2007). Image Time-Series Data Mining Based on the Information-Bottleneck Principle. *IEEE Trans. Geosci. Remote Sens.*, **45**, 827–838. 20

Gueguen, L. & Datcu, M. (2008). A Similarity Metric for Retrieval of Compressed Objects: Application for Mining Satellite Image Time Series. *IEEE Trans. Knowl. Data Eng.*, **20**, 562–575. 20

Gueguen, L. & Datcu, M. (2009). Mixed Information Measure: Application to Change Detection in Earth Observation. In *Proc. 5th International Workshop on the Analysis of Multi-temporal Remote Sensing Images, MultiTemp*, Connecticut, USA. 16, 45

Gueguen, L., Pesaresi, M., Ehrlich, D. & Lu, L. (2011a). Urbanization analysis by mutual information based change detection between SPOT 5 panchromatic images. In *Proc. 6th International Workshop on the Analysis of Multi-temporal Remote Sensing Images, MultiTemp*, 157–160, Trento, Italy. 16

Gueguen, L., Soille, P. & Pesaresi, M. (2011b). Change Detection Based on Information Measure. *IEEE Trans. Geosci. Remote Sens.*, **49**, 4503–4515. 16, 45, 51

Hall, F.G., Botkin, D.B., Strebel, D.E., Woods, K.D. & Goetz, S.J. (1991). Large scale patterns of forest succession as determined by remote sensing. *Ecology*, **72**, 628–640. 15

Haralick, R.M., Shanmugam, K. & Dinstein, I. (1973). Textural Features for Image Classification. *IEEE Trans. Syst., Man, Cybern.*, **3**, 610–621. 17, 73, 102

Heas, P. & Datcu, M. (2005). Modeling trajectory of dynamic clusters in image time-series for spatio-temporal reasoning. *IEEE Trans. Geosci. Remote Sens.*, **43**, 1635–1647. 20

Hershey, J.R. & Olsen, P.A. (2007). Approximating the Kullback Leibler Divergence Between Gaussian Mixture Models. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, ICASSP*, vol. 4, IV–317–IV–320, Honolulu, HawaII, USA. 44, 50

Huo, C., Zhou, Z., Lu, H., Pan, C. & Chen, K. (2010). Fast Object-Level Change Detection for VHR Images. *IEEE Geosci. Remote Sens. Lett.*, **7**, 118–122. 15

Inglada, J. & Giros, A. (2004). On the possibility of automatic multisensor image registration. *IEEE Trans. Geosci. Remote Sens.*, **42**, 2104–2120. 16

Inglada, J. & Mercier, G. (2007). A New Statistical Similarity Measure for Change Detection in Multitemporal SAR Images and Its Extension to Multiscale Change Analysis. *IEEE Trans. Geosci. Remote Sens.*, **45**, 1432–1445. 16, 50, 54, 66

Jacobson, N.P., Gupta, M.R. & Cole, J.B. (2007). Linear Fusion of Image Sets for Display. *IEEE Trans. Geosci. Remote Sens.*, **45**, 3277–3288. 23

Jakeman, E. (1980). On the statistics of K-distributed noise. *Journal of Physics A: Mathematical and General*, **13**, 31. 14

Jeon, B. & Landgrebe, D.A. (1992). Classification with spatio-temporal interpixel class dependency contexts. *IEEE Trans. Geosci. Remote Sens.*, **30**, 663–672. 15

Johnson, N.B., Norman Lloyd; Samuel Kotz (1995). *Continuous Univariate Distributions, Volume 2 (Second Edition, Section 27)*. Wiley. 30

Jonsson, P. & Eklundh, L. (2002). Seasonality extraction by function fitting to time-series of satellite sensor data. *IEEE Trans. Geosci. Remote Sens.*, **40**, 1824–1832. 21

Julea, A., Meger, N., Bolon, P., Rigotti, C., Doin, M.P., Lasserre, C., Trouve, E. & lazarescu, V. (2011). Unsupervised Spatiotemporal Mining of Satellite Image Time Series Using Grouped Frequent Sequential Patterns. *IEEE Trans. Geosci. Remote Sens.*, **49**, 1417–1430. 21

Julesz, B. (1981). Textons, the elements of texture perception, and their interactions. *Nature*, **290**, 91–97. 17

Kadir, T. & Brady, M. (2001). Saliency, Scale and Image Description. *Int. J. Comput. Vision*, **45**, 83–105. 43, 82

Kanzow, C., Yamashita, N. & Fukushima, M. (2005). Levenberg-Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints. *Journal of Computational and Applied Mathematics*, **173**, 321–343. 35

Karvonen, J. & Simila, M. (2002). A wavelet transform coder supporting browsing and transmission of sea ice SAR imagery. *IEEE Trans. Geosci. Remote Sens.*, **40**, 2464–2485. 17

Kern, J.P. & Pattichis, M.S. (2007). Robust Multispectral Image Registration Using Mutual-Information Models. *IEEE Trans. Geosci. Remote Sens.*, **45**, 1494–1505. 16

Kleynhans, W., Olivier, J.C., Wessels, K.J., van den Bergh, F., Salmon, B.P. & Steenkamp, K.C. (2010). Improving Land Cover Class Separation Using an Extended Kalman Filter on MODIS NDVI Time-Series Data. *IEEE Geosci. Remote Sens. Lett.*, **7**, 381–385. 21

Kotwal, K. & Chaudhuri, S. (2012). An Optimization-Based Approach to Fusion of Hyperspectral Images. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 501–509. 23

Kraskov, A., Stögbauer, H. & Grassberger, P. (2004). Estimating mutual information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, **69**. 45

Krylov, V. & Zerubia, J. (2010). Generalized gamma mixtures for supervised SAR image classification. In *GraphiCon*, 107–110, Saint-Petersburg, Russia. 30

Krylov, V.A., Moser, G., Serpico, S.B. & Zerubia, J. (2011). Supervised High-Resolution Dual-Polarization SAR Image Classification by Finite Mixtures and Copulas. *IEEE J. Sel. Topics Signal Process.*, **5**, 554–566. 13

Kuhn, H.W. & Tucker, A.W. (1950). Nonlinear Programming. In J. Neyman, ed., *Proceedings of the 2nd Berkeley Symposium on Mathematical Statistics and Probability*, 481–492, University of California Press, Berkeley, CA, USA. 116

Kuruoglu, E.E. & Zerubia, J. (2004). Modeling SAR images with a generalization of the Rayleigh distribution. *IEEE Trans. Image Process.*, **13**, 527–533. 14, 27

Lazebnik, S. & Raginsky, M. (2009). Supervised Learning of Quantizer Codebooks by Information Loss Minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, **31**, 1294 –1309. 87

Lazebnik, S., Schmid, C. & Ponce, J. (2005). A Sparse Texture Representation Using Local Affine Regions. *IEEE Trans. Pattern Anal. Mach. Intell.*, **27**, 1265–1278. 19, 83

Lazebnik, S., Schmid, C. & Ponce, J. (2006). Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *Proc IEEE Conference on Computer Vision and Pattern Recognition CVPR*, 2169–2178, Washington, DC, USA. 19, 86

Lazic, N. & Aarabi, P. (2007). Importance of Feature Locations in Bag-of-Words Image Classification. In *Proc IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, I–641–I–644, Honolulu, Hawaii, USA. 19

Le Moan, S., Mansouri, A., Voisin, Y. & Hardeberg, J.Y. (2011). A Constrained Band Selection Method Based on Information Measures for Spectral Image Color Visualization. *IEEE Trans. Geosci. Remote Sens.*, **49**, 5104–5115. 23

Lee, H. (2008). Mapping Deforestation and Age of Evergreen Trees by Applying a Binary Coding Method to Time-Series Landsat November Images. *IEEE Trans. Geosci. Remote Sens.*, **46**, 3926–3936. 21

Leung, T. & Malik, J. (2001). Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. *Int. J. Comput. Vision*, **43**, 29–44. 17, 18

Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quart. Appl. Math.*, **2**, 164–168. 35

Li, H.C., Hong, W. & Wu, Y.R. (2007). Generalized gamma distribution with MoLC estimation for statistical modeling of SAR images. In *Proc. 1st Asian and Pacific Conf. Synthetic Aperture Radar APSAR 2007*, 525–528, Anhui, China. 14, 30

Li, H.C., Hong, W., Wu, Y.R. & Fan, P.Z. (2010). An Efficient and Flexible Statistical Model Based on Generalized Gamma Distribution for Amplitude SAR Images. *IEEE Trans. Geosci. Remote Sens.*, **48**, 2711–2722. 14, 26, 27

Li, H.C., Hong, W., Wu, Y.R. & Fan, P.Z. (2011). On the Empirical-Statistical Modeling of SAR Images With Generalized Gamma Distribution. *IEEE J. Sel. Topics Signal Process.*, **5**, 386–397. 28, 30

Li, M., Chen, X., Li, X., Ma, B. & Vitanyi, P.M.B. (2004). The similarity metric. *IEEE Trans. Inf. Theory*, **50**, 3250–3264. 20

Li, Y. & Maguire, L. (2011). Selecting Critical Patterns Based on Local Geometrical and Statistical Information. *IEEE Trans. Pattern Anal. Mach. Intell.*, **33**, 1189–1201. 122

Lienou, M., Maitre, H. & Datcu, M. (2010). Semantic Annotation of Satellite Images Using Latent Dirichlet Allocation. *IEEE Geosci. Remote Sens. Lett.*, **7**, 28–32. 19

Liu, G., Jia, H., Zhang, R., Zhang, H., Jia, H., Yu, B. & Sang, M. (2011a). Exploration of Subsidence Estimation by Persistent Scatterer InSAR on Time Series of High Resolution TerraSAR-X Images. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 159–170. 22

Liu, H., Song, D., Rüger, S.M., Hu, R. & Uren, V.S. (2008). Comparing Dissimilarity Measures for Content-Based Image Retrieval. In *Proc. 4th Asia Information Retrieval Conference on Information Retrieval Technology*, AIRS'08, 44–50, Springer-Verlag, Harbin, China. 16

Liu, L. & Fieguth, P. (2012). Texture Classification from Random Features. *IEEE Trans. Pattern Anal. Mach. Intell.*, **34**, 574 –586. 83

Liu, L., Fieguth, P., Kuang, G. & Zha, H. (2011b). Sorted Random Projections for robust texture classification. In *Proc. IEEE International Conference on Computer Vision, ICCV*, 391–398, Barcelona, Spain. 83, 87, 92

Liu, L., Fieguth, P., Clausi, D. & Kuang, G. (2012). Sorted random projections for robust rotation-invariant texture classification. *Pattern Recogn.*, **45**, 2405–2418. 83, 87

Longbotham, N., Pacifici, F., Glenn, T., Zare, A., Volpi, M., Tuia, D., Christophe, E., Michel, J., Inglada, J., Chanussot, J. & Du, Q. (2012). Multi-Modal Change Detection, Application to the Detection of Flooded Areas: Outcome of the 2009–2010 Data Fusion Contest. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 331–342. 15

Lopes, A., Laur, H. & Nezry, E. (1990). Statistical Distribution And Texture In Multilook And Complex SAR Images. In *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2427–2430, Maryland, USA. 30

Lopez-Sanchez, J.M., Ballester-Berman, J.D. & Hajnsek, I. (2011). First Results of Rice Monitoring Practices in Spain by Means of Time Series of TerraSAR-X Dual-Pol Images. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 412–422. 8, 21

Lowe, D.G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, **60**, 91–110. 18

Madsen, K., Nielsen, H.B. & Tingleff, O. (2004). Methods for non-linear least squares problems. Technical University of Denmark, Lecture notes. 34

Mairal, J., Bach, F., Ponce, J., Sapiro, G. & Zisserman, A. (2008). Supervised Dictionary Learning. In *Proc. Advances in Neural Information Processing Systems, NIPS*. 87

Maji, S., Berg, A.C. & Malik, J. (2013). Efficient Classification for Additive Kernel SVMs. *IEEE Trans. Pattern Anal. Mach. Intell.*, **35**, 66–77. 117

Mallat, S.G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, **11**, 674–693. 62

Manjunath, B.S. & Ma, W.Y. (1996). Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell.*, **18**, 837–842. 17, 73, 80, 102

Marcelja, S. (1980). Mathematical description of the responses of simple cortical cells. *J. Opt. Soc. Am*, **70**, 1297–1300. 17

Maree, R., Geurts, P., Piater, J. & Wehenkel, L. (2005). Random Subwindows for Robust Image Classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, CVPR '05*, vol. 1, 34–40, Los Alamitos, CA, USA. 19, 82

Marquardt, D.W. (1963). An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM Journal on Applied Mathematics*, **11**, 431–441. 35

Marques, R.C.P., Medeiros, F.N. & Santos Nobre, J. (2012). SAR image segmentation based on level set approach and $\mathcal{G}_a^0$ model. *IEEE Trans. Pattern Anal. Mach. Intell.*, **34**, 2046–2057. 13

McCloy, K.R. & Lucht, W. (2004). Comparative evaluation of seasonal patterns in long time series of satellite image data and simulations of a global vegetation model. *IEEE Trans. Geosci. Remote Sens.*, **42**, 140–153. 20

Meila, M. (2003). Comparing Clusterings by the Variation of Information. In B. Schoelkopf & M.K. Warmuth, eds., *Learning Theory and Kernel Machines*, vol. 2777 of *Lecture Notes in Computer Science*, 173–187, Springer Berlin Heidelberg. 45

Meisner, R.E., Bittner, M. & Dech, S.W. (1999). Computer animation of remote sensing-based time series data sets. *IEEE Trans. Geosci. Remote Sens.*, **37**, 1100–1106. 23

Mercier, G. & Inglada, J. (2008). Change detection with misregistration errors and heterogeneous data through the Orfeo toolbox. Tech. rep., Institut TELECOM, TELECOM Bretagne. 16

Mercier, G., Derrode, S., Pieczynski, W., Nicolas, J.M., Joannic-Chardin, A. & Inglada, J. (2006). Copula-based Stochastic Kernels for Abrupt Change Detection. In *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 204–207, Denver, Colorado, USA. 15, 16

Michelle L. Bell, J.M.S., Francesca Dominici (2005). A meta-analysis of time-series studies of ozone and mortality with comparison to the national morbidity, mortality, and air pollution study. *Epidemiology*, **16**, 436–445. 20

Mikolajczyk, K. & Schmid, C. (2004). Scale & Affine Invariant Interest Point Detectors. *Int. J. Comput. Vision*, **60**, 63–86. 82

Mikolajczyk, K. & Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, **27**, 1615–1630. 82

Moosmann, F., Triggs, B. & Jurie, F. (2006). Fast Discriminative Visual Codebooks using Randomized Clustering Forests. In *Advances in Neural Information Processing Systems 19*, 985–992. 84

Moser, G. & Serpico, S.B. (2006). Generalized minimum-error thresholding for unsupervised change detection from SAR amplitude imagery. *IEEE Trans. Geosci. Remote Sens.*, **44**, 2972–2982. 13

Moser, G., Zerubia, J. & Serpico, S.B. (2006). SAR amplitude probability density function estimation based on a generalized gaussian model. *IEEE Trans. Image Process.*, **15**, 1429–1442. 14, 27

Nicolas, J.M. (2002). Introduction to second kind statistics: Application of log-moments and log-cumulants to analysis of radar images. *Traitement du Signal*, **19**, 139–167. 14, 31, 33

Nister, D. & Stewenius, H. (2006). Scalable Recognition with a Vocabulary Tree. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2161–2168, Washington, DC, USA. 84

Notarnicola, C., Ratti, R., Maddalena, V., Schellenberger, T., Ventura, B. & Zebisch, M. (2013). Seasonal Snow Cover Mapping in Alpine Areas Through Time Series of COSMO-SkyMed Images. *IEEE Geosci. Remote Sens. Lett.*, **10**, 716–720. 21

Nowak, E., Jurie, F. & Triggs, B. (2006). Sampling Strategies for Bag-of-Features Image Classification. In A. Leonardis, H. Bischof & A. Pinz, eds., *Proc. 7th European Conference on Computer Vision, ECCV*, vol. 3954 of *Lecture Notes in Computer Science*, 490–503, Springer Berlin Heidelberg. 19, 82

Ojala, T., Pietikäinen, M. & Mäenpää, T. (2002). Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, **24**, 971–987. 19

Oliver, C.J. (1993). Optimum texture estimators for SAR clutter. *J. Phys. D: Appl. Phys.*, **26**, 1824–1835. 14, 30

Oliver, C.J. & Quegan, S. (1998). *Understanding Synthetic Aperture Radar Images*. Nrowood, MA: Artech House. 14, 26, 30

Omidvari, M., Hassanzadeh, S. & Hosseinibalam, F. (2008). Time series analysis of ozone data in Isfahan. *Physica A: Statistical Mechanics and its Applications*, **387**, 4393 – 4403. 20

Paget, R. & Longstaff, I.D. (1998). Texture synthesis via a noncausal nonparametric multiscale Markov random field. *IEEE Trans. Image Process.*, **7**, 925–931. 49

Palomar, D.P. & Verdu, S. (2008). Lautum Information. *IEEE Trans. Inf. Theory*, **54**, 964–975. 45

Patané, G. & Russo, M. (2001). The enhanced LBG algorithm. *Neural Networks*, **14**, 1219–1237. 80

Perronnin, F. (2008). Universal and Adapted Vocabularies for Generic Visual Categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, **30**, 1243–1256. 87

Perronnin, F., Sánchez, J. & Mensink, T. (2010). Improving the Fisher Kernel for Large-Scale Image Classification. In *Proc. 11th European Conference on Computer Vision, ECCV*, 143–156, Springer-Verlag, Heraklion, Crete, Greece. 85, 100

Petitjean, F., Inglada, J. & Gancarski, P. (2012). Satellite Image Time Series Analysis Under Time Warping. *IEEE Trans. Geosci. Remote Sens.*, **50**, 3081–3095. 21

Philbin, J., Chum, O., Isard, M., Sivic, J. & Zisserman, A. (2007). Object retrieval with large vocabularies and fast spatial matching. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 1–8, Minneapolis, Minnesota, USA. 84

Platt, J.C. (2000). Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In *Advances in Large Margin Classifiers*, 61–74. 118

Popescu, A., Gavat, I. & Datcu, M. (2008). Complex SAR image characterization using space variant spectral analysis. In *Proc. IEEE Radar Conference, RADAR '08*, 1–4, Rome, Italy. 102

Popescu, A.A., Gavat, I. & Datcu, M. (2012). Contextual Descriptors for Scene Classes in Very High Resolution SAR Images. *IEEE Geosci. Remote Sens. Lett.*, **9**, 80–84. 22, 74

Quegan, S., Le Toan, T., Yu, J.J., Ribbes, F. & Floury, N. (2000). Multitemporal ERS SAR analysis applied to forest mapping. *IEEE Trans. Geosci. Remote Sens.*, **38**, 741–753. 20

Rignot, E.J.M. & van Zyl, J.J. (1993). Change detection techniques for ERS-1 SAR data. *IEEE Trans. Geosci. Remote Sens.*, **31**, 896–906. 15

Romani, L.A.S., de Avila, A.M.H., Chino, D.Y.T., Zullo, J., Chbeir, R., Traina, C. & Traina, A.J.M. (2013). A New Time Series Mining Approach Applied to Multitemporal Remote Sensing Imagery. *IEEE Trans. Geosci. Remote Sens.*, **51**, 140–150. 21

Roweis, S.T. & Saul, L.K. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, **290**, 2323–2326. 23

S. Le HÃĽgarat-Mascle, R.S. (2004). Automatic change detection by evidential fusion of change indices. *Remote Sensing of Environment*, **91**, 390–404. 16

Salmon, B.P., Olivier, J.C., Wessels, K.J., Kleynhans, W., van den Bergh, F. & Steenkamp, K.C. (2011). Unsupervised Land Cover Change Detection: Meaningful Sequential Time Series Analysis. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 327–335. 15, 21

Satellites (2014). Satellite instruments. `http://database.eohandbook.com/database/instrumenttable.aspx`. 21

Selesnick, I., Baraniuk, R. & Kingsbury, N. (2005). The dual-tree complex wavelet transform. *IEEE Signal Process. Mag.*, **22**, 123 – 151. 102

Settles, B. (2009). Active Learning Literature Survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison. 112, 119

Shannon, C.E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, **27**, 379–423. 25

Sharifi, K. & Leon-Garcia, A. (1995). Estimation of shape parameter for generalized Gaussian distributions in subband decompositions of video. *IEEE Trans. Circuits Syst. Video Technol.*, **5**, 52–56. 62

Shimazaki, H. & Shinomoto, S. (2010). Kernel bandwidth optimization in spike rate estimation. *Journal of Computational Neuroscience*, **29**, 171–182. 38

Shyu, C.R., Klaric, M., Scott, G.J., Barb, A.S., Davis, C.H. & Palaniappan, K. (2007). GeoIRIS: Geospatial information retrieval and indexing system — content mining, semantics modeling, and complex queries. *IEEE Trans. Geosci. Remote Sens.*, **45**, 839–852. 22, 74

Simoncelli, E. & Adelson, E. (1990). Non-separable extensions of quadrature mirror filters to multiple dimensions. In *Proceedings of the IEEE*, vol. 78, 652–664. 102

Singh, A. (1989). Digital change detection techniques using remotely-sensed data. *Int. J. Remote Sens.*, **10**, 989–1003. 15

Singh, J. & Datcu, M. (2012). Mining very high resolution complex-valued SAR images using the fractional Fourier transform. In *Proc. 9th European Conference on Synthetic Aperture Radar, EUSAR*, 135–138, Nuernberg, Germany. 102

Singh, J. & Datcu, M. (2013). SAR Image Categorization With Log Cumulants of the Fractional Fourier Transform Coefficients. *IEEE Trans. Geosci. Remote Sens.*, **51**, 5273–5282, early Access. 102

Singh, J., Cui, S., Datcu, M. & Gleich, D. (2012). A survey of density estimation for SAR images. In *Proc. 20th European Signal Processing Conference, EUSIPCO*, 2526–2530, Bucharest, Romania. 41

Sivic, J. & Zisserman, A. (2003). Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proc. IEEE International Conference on Computer Vision, ICCV*, vol. 2, 1470–1477, Washington, DC, USA. 18, 100

Soergel, U., ed. (2010). *Radar Remote Sensing of Urban Areas*. Springer Netherland. 1

Solberg, A.H.S., Taxt, T. & Jain, A.K. (1996). A Markov random field model for classification of multisource satellite imagery. *IEEE Trans. Geosci. Remote Sens.*, **34**, 100–113. 15

Song, K.S. (2006). A globally convergent and consistent method for estimating the shape parameter of a generalized Gaussian distribution. *IEEE Trans. Inf. Theory*, **52**, 510–527. 62

Song, K.S. (2008). Globally Convergent Algorithms for Estimating Generalized Gamma Distributions in Fast Signal and Image Processing. *IEEE Trans. Image Process.*, **17**, 1233–1250. 63

Strang, G. & Nguyen, T.Q. (1997). *Wavelets and filter banks*. Wellesley-Cambridge Press. 65, 102

Sun, H., Sun, X., Wang, H., Li, Y. & Li, X. (2012). Automatic Target Detection in High-Resolution Remote Sensing Images Using Spatial Sparse Coding Bag-of-Words Model. *IEEE Geosci. Remote Sens. Lett.*, **9**, 109–113. 19

Suri, S. & Reinartz, P. (2010). Mutual-Information-Based Registration of TerraSAR-X and Ikonos Imagery in Urban Areas. *IEEE Trans. Geosci. Remote Sens.*, **48**, 939–949. 16

Tenenbaum, J.B. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, **290**, 2319–2323. 23

Tison, C., Nicolas, J.M., Tupin, F. & Maitre, H. (2004). A new statistical model for Markovian classification of urban areas in high-resolution SAR images. *IEEE Trans. Geosci. Remote Sens.*, **42**, 2046–2057. 13, 30, 56

Tong, S. (2001). *Active learning: theory and applications*. Ph.D. thesis, Stanford University, Available from: http://www.robotics.stanford.edu/~stong/research.html, [Accessed 8 November 2013]. 119

Tourassi, G.D., Harrawood, B., Singh, S., Lo, J.Y. & Floyd, C.E. (2007). Evaluation of information-theoretic similarity measures for content-based retrieval and detection of masses in mammograms. *Medical Physics*, **34**, 140–150. 16, 44

Touzi, R., Lopes, A. & Bousquet, P. (1988). A statistical and geometrical edge detector for SAR images. *IEEE Trans. Geosci. Remote Sens.*, **26**, 764–773. 13, 73, 74

Tuia, D., Volpi, M., Copa, L., Kanevski, M. & Munoz-Mari, J. (2011). A Survey of Active Learning Algorithms for Supervised Remote Sensing Image Classification. *IEEE J. Sel. Topics Signal Process.*, **5**, 606 –617. 22

Tyo, J.S., Konsolakis, A., Diersen, D.I. & Olsen, R.C. (2003). Principal-components-based display strategy for spectral imagery. *IEEE Trans. Geosci. Remote Sens.*, **41**, 708–718. 23

Udelhoven, T. (2011). TimeStats: A Software Tool for the Retrieval of Temporal Patterns From Global Satellite Archives. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 310–317. 21

Ulaby, F.T., Kouyate, F., Brisco, B. & Williams, T.H.L. (1986). Textural Information in SAR Images. *IEEE Trans. Geosci. Remote Sens.*, **24**, 235–245. 30

Vaduva, C., Costa chioiu, T., T. a chioiu, Pa tras, C., cu, Gava t, I., L a za rescu, V. & Datcu, M. (2013). A Latent Analysis of Earth Surface Dynamic Evolution Using Change Map Time Series. *IEEE Trans. Geosci. Remote Sens.*, **51**, 2105–2118. 21

Van de Wouwer, G., Scheunders, P. & Van Dyck, D. (1999). Statistical texture characterization from discrete wavelet representations. *IEEE Trans. Image Process.*, **8**, 592–598. 63

van Gemert, J.C., Veenman, C.J., Smeulders, A.W.M. & Geusebroek, J.M. (2010). Visual Word Ambiguity. *IEEE Trans. Pattern Anal. Mach. Intell.*, **32**, 1271–1283. 20, 76, 85, 100

Varma, M. & Zisserman, A. (2005). A Statistical Approach to Texture Classification from Single Images. *Int. J. Comput. Vision*, **62**, 61–81. 17, 18, 83

Varma, M. & Zisserman, A. (2009). A Statistical Approach to Material Classification Using Image Patch Exemplars. *IEEE Trans. Pattern Anal. Mach. Intell.*, **31**, 2032–2047. 83, 84

Viola, P. & Jones, M.J. (2004). Robust Real-Time Face Detection. *Int. J. Comput. Vision*, **57**, 137–154. 22, 114

Viovy, N. & Saint, G. (1994). Hidden Markov models applied to vegetation dynamics analysis using satellite remote sensing. *IEEE Trans. Geosci. Remote Sens.*, **32**, 906–917. 20

Vlassis, N.A. & Likas, A. (2002). A Greedy EM Algorithm for Gaussian Mixture Learning. *Neural Processing Letters*, **15**, 77–87. 37

Vogelmann, J.E., Kost, J.R., Tolk, B., Howard, S., Short, K., Chen, X., Huang, C., Pabst, K. & Rollins, M.G. (2011). Monitoring Landscape Change for LANDFIRE Using Multi-Temporal Satellite Imagery and Ancillary Data. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **4**, 252–264. 21

Voisin, A., Krylov, V.A., Moser, G., Serpico, S.B. & Zerubia, J. (2013). Classification of Very High Resolution SAR Images of Urban Areas Using Copulas and Texture in a Hierarchical Markov Random Field Model. *IEEE Geosci. Remote Sens. Lett.*, **10**, 96–100. 13

Walessa, M. & Datcu, M. (2000). Model-based despeckling and information extraction from SAR images. *IEEE Trans. Geosci. Remote Sens.*, **38**, 2258–2269. 17

Wang, H.Q., Li, H.C. & Huang, P.P. (2012). Change detection based GGR-GKIT on SAR amplitude image. In *Proc. 9th European Conference on Synthetic Aperture Radar, EUSAR*, 199–202, Nuernberg, Germany. 13

Wang, J., Yang, J., Yu, K., Lv, F., Huang, T. & Gong, Y. (2010). Locality-constrained Linear Coding for Image Classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition CVPR*, 3360–3367, San Francisco, CA, USA. 85, 86, 100

Wang, J., Li, Y., Zhang, Y., Wang, C., Xie, H., Chen, G. & Gao, X. (2011). Bag-of-Features Based Medical Image Retrieval via Multiple Assignment and Visual Words Weighting. *IEEE Trans. Med. Imag.*, **30**, 1996–2011. 80

Ward, K.D. (1981). Compound representation of high resolution sea clutter. *Electronics Letters*, **17**, 561–563. 13, 28

William, H., Teukolsky, S.A., Vetterling, W.T. & Flannery, B.P. (2007). *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3rd edn. 38

Wingo, D.R. (1987). Computing Maximum-Likelihood Parameter Estimates of the Generalized Gamma Distribution by Numerical Root Isolation. *IEEE Trans. Rel.*, **36**, 586–590. 32

Winter, A., Maitre, H., Cambou, N. & Legrand, E. (1997). Entropy and multiscale analysis: a new feature extraction algorithm for aerial images. In *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing ICASSP*, vol. 4, 2765–2768, Munich, Germany. 16

Wu, J. & Rehg, J.M. (2011). CENTRIST: A Visual Descriptor for Scene Categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, **33**, 1489–1501. 19

Wu, J., Brubaker, S.C., Mullin, M.D. & Rehg, J.M. (2008). Fast Asymmetric Learning for Cascade Face Detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, **30**, 369–382. 114

Xu, S., Fang, T., Li, D. & Wang, S. (2010). Object Classification of Aerial Images With Bag-of-Visual Words. *IEEE Geosci. Remote Sens. Lett.*, **7**, 366–370. 19, 80

Yan, Y., Doin, M.P., Lopez-Quiroz, P., Tupin, F., Fruneau, B., Pinel, V. & Trouve, E. (2012). Mexico City Subsidence Measured by InSAR Time Series: Joint Analysis Using PS and SBAS Approaches. *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.*, **5**, 1312–1326. 22

Yang, J., Yu, K., Gong, Y. & Huang, T.S. (2009). Linear spatial pyramid matching using sparse coding for image classification. In *Proc IEEE Conference on Computer Vision and Pattern Recognition CVPR*, 1794–1801, Miami, Florida, USA. 19, 86

Yang, Y. & Newsam, S. (2010). Bag-of-visual-words and spatial extensions for land-use classification. In *Proc. 18th International Conference on Advances in Geographic Information Systems*, GIS '10, 270–279, San Jose, California. 19

Yu, K., Zhang, T. & Gong, Y. (2009). Nonlinear Learning using Local Coordinate Coding. In *Advances in Neural Information Processing Systems 22*, 2223–2231. 86

Zhao, G., Ahonen, T., Matas, J. & Pietikainen, M. (2012). Rotation-Invariant Image and Video Description With Local Binary Pattern Features. *IEEE Trans. Image Process.*, **21**, 1465 –1477. 19

Zheng, J. & You, H. (2013). A New Model-Independent Method for Change Detection in Multitemporal SAR Images Based on Radon Transform and Jeffrey Divergence. *IEEE Geosci. Remote Sens. Lett.*, **10**, 91–95. 59

Zhu, S.C., Guo, C.E., Wang, Y. & Xu, Z. (2005). What are Textons? *Int. J. Comput. Vision*, **62**, 121–143. 17

Zhu, X., Goldberg, A.B., Brachman, R. & Dietterich, T. (2009). *Introduction to Semi-Supervised Learning*. Morgan and Claypool Publishers. 22