Semi-supervised Hierarchical Clustering for Semantic SAR Image Annotation

Wei Yao, Corneliu Octavian Dumitru, Otmar Loffeld, *Senior Member, IEEE*, and Mihai Datcu, *Fellow, IEEE*

***Abstract*—In this paper, we propose a semi-automated hierarchical clustering and classification framework for synthetic aperture radar (SAR) image annotation. Our implementation of the framework allows the classification and annotation of image data ranging from scenes up to large satellite data archives. Our framework comprises three stages: 1) each image is cut into patches and each patch is transformed into a texture feature vector; 2) similar feature vectors are grouped into clusters, where the number of clusters is determined by repeated cluster splitting to optimize their Gaussianity; and 3) the most appropriate class (i.e., a semantic label) is assigned to each image patch. This is accomplished by semi-supervised learning. For the testing and validation of our implemented framework, a concept for a two-level hierarchical semantic image content annotation was designed and applied to a manually annotated reference dataset consisting of various TerraSAR-X image patches with meter-scale resolution. Here, the upper level contains general classes, while the lower level provides more detailed subclasses for each parent class. For a quantitative and visual evaluation of the proposed framework, we compared the relationships among the clustering results, the semi-supervised classification results, and the two-level annotations. It turned out that our proposed method is able to obtain reliable results for the upper-level (i.e., general class) semantic classes; however, due to the too many detailed subclasses versus the few instances of each subclass, the proposed method generates inferior results for the lower level. The most important contributions of this paper are the integration of modified Gaussian-means and modified cluster-then-label algorithms, for the purpose of large-scale SAR image annotation, as well as the measurement of the clustering and classification performances of various distance metrics.**

***Index Terms*—Gaussian hypothesis test, hierarchical clustering, semantic annotation, semi-supervision, similarity measures.**

I. Introduction

A. Introduction and State-of-the-Art

With the increased availability of Earth observation (EO) data, due to new satellite missions, their various sensors, and the interoperability of data archives, the remote sensing community is facing today a dramatic increase in both data volume and data content levels.

Manuscript received June 18, 2015; revised October 09, 2015, February 01, 2016; accepted February 24, 2016. Date of publication April 07, 2016; date of current version April 22, 2016. This work was supported by the MOSES (Multi-Modal Sensor Systems for Environmental Exploration) postgraduate programme.

W. Yao and O. Loffeld are with the Center for Sensor Systems (ZESS), University of Siegen, Siegen 57076, Germany (e-mail: yao@zess.uni-siegen.de; wei.yao@dlr.de; loffeld@zess.uni-siegen.de).

C. O. Dumitru and M. Datcu are with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, Wessling 82230, Germany (e-mail: corneliu.dumitru@dlr.de; mihai.datcu@dlr.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTARS.2016.2537548

Many EO applications stem from the early days of the geoscience and remote sensing discipline, such as measuring land-use characteristics, monitoring and responding to natural disasters, managing natural resources, etc. However, with regard to the data volume as well as the data content, the current abundant satellite data are far from being well processed, analyzed, and utilized.

Compared to computer vision applications that mainly focus on ordinary optical images which cover very detailed objects within a small field of view, remote sensing images are often used for EO purposes covering large areas that contain rich details. Nowadays, computer vision experts are making efforts for understanding parts of images that we probably neglect at first glance [1]. They have collected many typical image datasets and their understanding is supported by image annotation. Most of the available semantically annotated image collections, which contain a large variety of retrievable objects and classes, have been built by individual groups with the intention to solve specific problems. Typical examples are the LabelMe [2] and SUN (scene understanding) datasets [3]. In contrast, remote sensing applications have not reached the same level of maturity yet.

Hence, one of the urgent but not yet well-solved tasks in remote sensing is the so-called image annotation in large-scale remote sensing datasets. The procedure of attaching text labels to elements of images explaining their content is called "semantic image annotation." Its purpose in remote sensing is to reach a better understanding not only of the built-up environment that we live in, but also of the natural environment that surrounds us.

However, it is well known that manual annotation requires very much human effort, because labeling an image forces us to become aware of the detailed image content which costs much time [1]. For fast browsing and indexing of image content in a large-scale image dataset, it is critical to develop techniques which are able to semantically annotate a given dataset efficiently and with high quality.

In this context, image annotation may be performed with supervision, without supervision, or in an "intermediate" approach. It includes supervised learning [4], unsupervised learning [5], [6], as well as the "intermediate" form of supervision, i.e., semi-supervised learning [7], or active learning [8].

Due to its excellent performance, latent Dirichlet allocation (LDA), a generative model, is often used for classification in natural language processing. The original algorithm and its variations have been introduced and extended into the remote sensing domain by [5], [9], and [10]. Similarly, inspired by the relationships among chromosomes, DNA, and genes in

biological systems, an image-to-concept distribution model has been proposed by [6] to obtain reliable semantic annotations.

Furthermore, as both supervised learning and unsupervised learning have their own strong advantages and disadvantages, a probabilistic formulation, which combines the advantages of the two methods via a reformulation of the supervised approach, has been proposed for semantic image annotation and retrieval by [4].

Moreover, a semi-supervised algorithm, which trains a hierarchical latent variable model with both labeled and unlabeled data, has been proposed for auto-annotation and unknown structure discovery in satellite images by [7], and a multiscale coarse-to-fine cascaded active learning method to retrieve patterns in large image datasets, has been proposed by [8].

Finally, as complex scenes are difficult to describe with a single label for each image patch, a hierarchical semantic multiinstance multilabel learning (MIML) framework for high-resolution remote sensing image annotation via a Gaussian process has been proposed by [11]. Some other applications even consider extra information for image annotation, e.g., visual contexts can be learned for image annotation based on Flickr group labels [12].

B. Methodology Concept

Due to the large data volume and the scene content complexity, the trend of remote sensing image annotation is going into the direction of an intermediate level of supervision, rather than to full or no supervision. In this article, we also follow this path by presenting a semi-supervised Gaussianity-based hierarchical clustering method for remote sensing image annotation. Specifically, our method intends to answer the following two questions: How do we explore information in the feature vector space, and how do we link our semi-supervised results with an already annotated reference dataset?

Hence, one goal of this research is to study the extracted features in their high-dimensional feature space, i.e., how do they behave (including their computational effort) and what is a good distance metric to describe the pair-wise relationships between the feature points. Therefore, the definition of distance metrics plays an important role in exploring features in a given feature space (that, as a rule, has more than three dimensions [13]).

The second goal is to find the relationships between the clustering results, the semi-supervised classification results, and the two-level annotations. When we group the feature points based on their similarity, we are able to generate homogeneous clusters. Then, a cluster-then-label semi-supervised learning method is performed [14].

In order to reach these two goals, we have to start with some exploratory data analysis to identify the main data characteristics, verify our statistical assumptions, select appropriate models, and determine the hidden relationships among the variables. For the specific application of synthetic aperture radar (SAR) imaging which uses successive pulses of microwaves to actively illuminate a scene, the processing performances have to be analyzed and evaluated thoroughly.

Then, we continue with the preparation and application of reference data that we need for the evaluation and validation of our method. This includes the performance analysis of clustering as well as classification, where we apply quantitative tests and subjective user assessments that may jeopardize the objectivity of the reference data and influence the evaluation results.

Currently, in the computer vision field, deep learning is the most popular technique [15]. However, it usually requires large datasets and long runtime for training.

In order to tackle the above problems, an unsupervised clustering method, which relies on feature vector similarity, is our preferred choice to explore the feature space. To become independent from the actual statistical distribution of the image data, we aim at a homogeneous clustering of feature vectors which allows us to group similar patches within the dataset as proposed by [16].

For a typical large satellite scene, it is preferable to represent the image content as a hierarchical structure with several general and detailed semantic content levels. For example, urban areas can be classified into subclasses such as densely built-up areas, and sparsely built-up areas. In hierarchical clustering, the main problem is to find a suitable criterion for cluster splitting. This will be further explained in Section II.

In addition, we have to tackle the "curse of dimensionality" problem. Therefore, the fractional, L1, L2, and Lp (Minkowski) distance metrics are studied in Section II, too. These distance metrics reduce multidimensional distances to simple scalars.

In the end, we compare the clustering results of the above-mentioned hierarchical clustering method and the cluster-then-label semi-supervised results with the reference data annotations in order to analyze the correspondences between the learning results and manually annotated reference data. The results are evaluated by visual and quantitative analysis. As known, the different understanding of the same image by a human and a machine is called "Semantic Gap" [17], [18].

Fig. 1 illustrates the whole concept of our proposed SAR data processing and analysis chain. We start from a semantically annotated dataset of image patches, from which we select a number of collections. Then, we perform feature extraction for each image patch. This results in a high-dimensional feature space. A clustering algorithm with a Gaussian test and distances metric are used to generate a dichotomized hierarchical cluster tree structure. After applying supervised learning in each cluster, predefined semantic labels are assigned to the feature vector points.

C. Dataset

Since its launch, the German TerraSAR-X EO satellite has already acquired thousands of high-resolution SAR images that have been processed into different levels of products. Their resolution is usually 1–5 m/pixel; thus, we can clearly recognize very detailed structures (e.g., industrial constructions, residential buildings) [19].

To solve typical remote sensing image analysis tasks, we focused on multilook ground detected and radiometrically enhanced high-resolution SpotLight mode products of

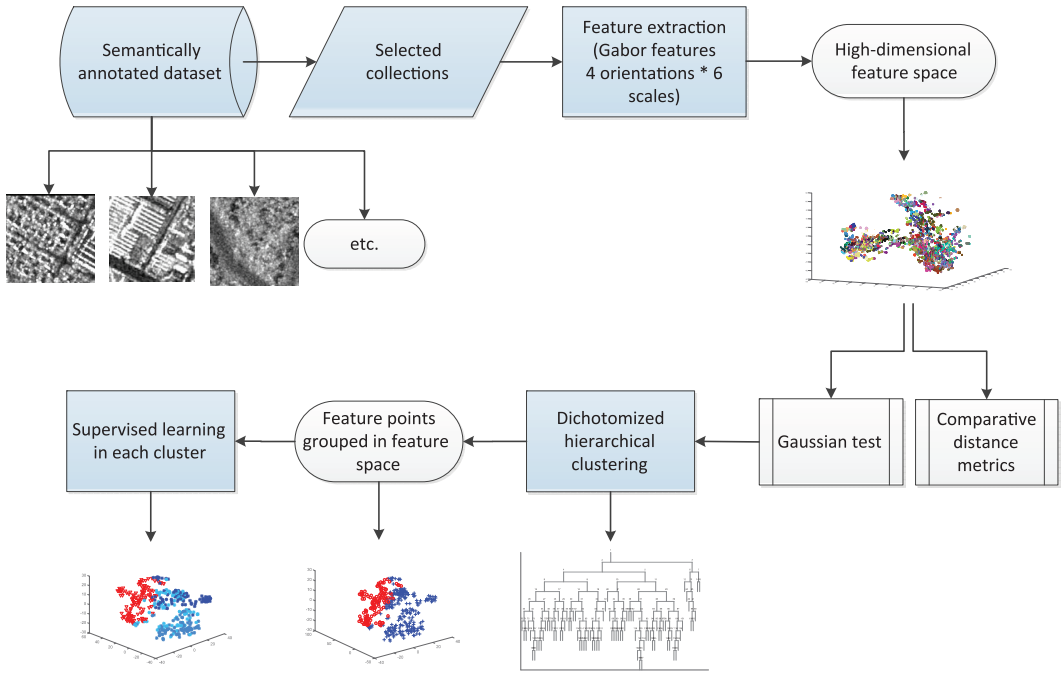


Fig. 1. Proposed SAR data processing and analysis chain.

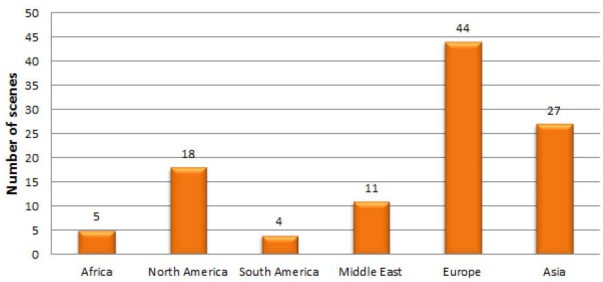


Fig. 2. Geographical distribution of our semantically annotated TerraSAR-X dataset.

TerraSAR-X. In this mode, the resolution of the TerraSAR-X images is about 2.9 m with a pixel spacing of 1.25 m. The average size of the given images is 4500×6500 pixels.

Our semantically annotated dataset consisted of more than 100 scenes covering different urban and nonurban areas around the world (see Fig. 2). Each image was tiled into patches of 160×160 pixels thus generating around 110 000 patches.

II. Methodology

As suggested in [16] and [20], feature clustering can be used as a basic component of an image retrieval system to find similar images and to exploit hidden information. Hence, in this section, a detailed description of our proposed method, which combines Gaussian-test-based hierarchical clustering and cluster-then-label semi-supervised learning, will be given.

A. Creation of a Reference Dataset

In this subsection, we present the methodology to classify and semantically annotate our TerraSAR-X images as a reference dataset. Our general approach during the annotation scheme was to tile each TerraSAR-X image into a number of nonoverlapping patches, extract features, then classify and annotate each individual patch. The main steps of the processing chain were:

1) Select TerraSAR-X images and tile into patches with a size of 160×160 pixels. Subsample each patch into a smaller patch of 80×80 pixels to generate decorrelated pixels [21].
2) Extract a 48-dimensional feature vector from each patch using Gabor filters with four scales and six orientations (take the mean and variance of the patch coefficients) [21].
3) Classify the feature vectors into classes using a support vector machine (SVM) with relevance feedback [22]. Each patch is assigned to a single class based on the dominant content of the patch.
4) Annotate each class by giving an appropriate semantic meaning to each class [23]. Google Earth is used as ground truth for visual support.

The annotation chain was semi-automated. The first two steps were automated, while the last two steps required manual interaction. For classification, an operator had to rank the given positive and negative examples and grouped them into classes of relevance; for annotation, the operator selected a proper semantic label for each class from a list of available labels. We defined a nomenclature adapted to TerraSAR-X images with a two-level hierarchical scheme that consists of a total of 150 semantic classes [24]. The upper-level semantic annotation contained eight general classes (settlements, industrial production areas, military facilities, transport, agriculture, natural vegetation, bare ground, and water bodies) that were later split into lower-level detailed subclasses. For example, agriculture was split into the following subclasses: cropland, stubble, bare land, ploughed agricultural land, rice paddies, pasture, plantations

and vegetables, greenhouses, and vineyards. Fig. 3 shows some examples of the annotated classes.

B. Hierarchical Clustering

In routine operations, the same Gabor features extraction steps, which was used to create the reference dataset, were executed for each newly acquired image to be analyzed; however, we had to adapt the feature clustering step.

Since, the classification performance of each reference class is unpredictable, it is difficult to expect a direct correspondence between clusters and classes. Each class can spread out into several clusters and we have to generate a hierarchical cluster structure to overcome this difficulty. Due to its simplicity and efficiency, k-means clustering can be used to split the initial clusters to construct a hierarchical structure. As a result, convex clusters are grouped in feature space and homogeneous clusters are obtained. A problem of k-means is to select the optimal number of clusters. This can be reached, however, by a proper splitting rule. Therefore, we use the Gaussian-means (G-means) algorithm proposed in [25]. The Central Limit Theory tells us that the arithmetic mean of a sufficiently large number of iterates of independent random variables, each with a well-defined expected value and a well-defined variance, will be approximately normally distributed, regardless of the underlying distribution.

Our splitting rule is to check whether a cluster follows a Gaussian distribution. Thus, there is no need to preset a fixed number of clusters. To prevent cluster generation with too few patches, we define an additional constraint for a minimum cluster size and we modify the G-means algorithm accordingly.

1) Modified G-Means Algorithm: The modified G-means algorithm starts with the entire dataset. During each iteration of the algorithm, a cluster, which does not follow a Gaussian distribution, is split into two new clusters, while all clusters that already obey a Gaussian distribution are preserved in the generated hierarchical structure. For computational efficiency, the Gaussian hypothesis test is simplified to process only an $n \times p$ dimensional feature data are projected onto the vector direction formed by the two new cluster centers. Thus, not the multivariate Gaussianity, but the univariate Gaussianity of a 1-D projection of the clusters is being tested. Because of the way Gaussianity testing is performed, a hierarchical clustering structure is naturally created.

Algorithm 1 describes our modified G-means algorithm.

The modified G-means algorithm is compared in the evaluation part for comparison with the original G-means algorithm. During implementation, we experiment with various distance metrics and a tree data structure is used to store the overall clustering structure. The tree structure not only enables us to observe how the entire dataset splits into smaller clusters at different levels, but also gives us the possibility to explore the relations among the general and detailed classes.

Algorithm 1. Modified G-means (X, α, $size$) algorithm

Data: Feature matrix $X_{n\times p}$, a confidence level α, and a minimum size constraint s.
Result: A hierarchical structure of clusters.
Initialize C, D, G, S; //the cluster centers, cluster data, Gaussianity booleans, cluster sizes
while !($g_i = 1$ or $s_i \leq s$) **do**
  Use k-means with k=2 to split cluster i;
  Project the feature matrix of cluster i onto the direction defined by the two newly generated cluster centers to obtain vector X';
  Use an Anderson-Darling test for vector X' under the confidence level α to check whether the 1D projection of cluster i follows a univariate Gaussian distribution;
  if *cluster i follows a Gaussian* **then**
    keep c_i;
    g_i = 1;
  else
    update C, D, G, S with new cluster parameters;
  end
end

C. Gaussian Hypothesis Testing

Based on the method proposed in [25], the Anderson–Darling Gaussian test performs better than the Bayesian information criterion (BIC) in finding a good stopping rule for cluster splitting. Hence, we use the Anderson–Darling Gaussian test to construct a hierarchical clustering, where an existing cluster is split into two new clusters when the null hypothesis is rejected, and the cluster splitting is stopped when the null hypothesis is accepted. The definition of the null hypothesis is shown below. The Gaussian hypothesis is chosen because of its ability in obtaining homogeneous clusters as well as its simplicity in implementation.

1) Feature Vector Projection: The commonly used Gaussian hypothesis test is suitable for data points along 1-D. In the case of high-dimensional data, the computational complexity increases dramatically. Therefore, we have to re-project the feature vectors to make them amenable to 1-D Gaussian hypothesis testing [25]

1) For a set of data points X with $n \times p$ dimensions, initialize two centers using the k-means++ algorithm [26], and run k-means on these two centers in X until convergence is reached.
2) The centers c_1, c_2 are obtained by k-means++. Let $v = c_1 - c_2$ be a p-element vector which connects the two centers. This is the cardinal direction k-means exploits for clustering.
3) Project all the data points X onto this preferred direction v, $x_i' = < x_i, v > / \|v\|^2$. X' is thus a 1-D representation of the data X projected onto v.
4) Normalize X' so that its mean value becomes 0 and its variance equals 1. The normalized X' is the projected feature vector which will be used and tested in the following.

The linear projection works as a reverse case of projection pursuit and independent component analysis (ICA). A linear projection of a random vector is a linear combination of its components. As indicated by the idea of ICA, indirectly due to the

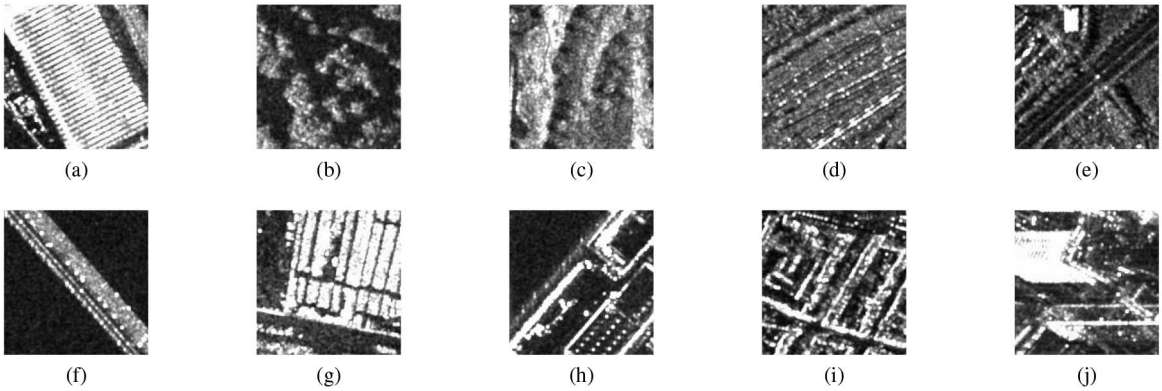


Fig. 3. Examples of annotated classes. (a) Industrial buildings. (b) Mixed forest. (c) Mountains. (d) Railways. (e) Roads. (f) Bridges. (g) Depots. (h) Harbor infrastructure. (i) High-density residential areas. (j) Skyscrapers.

central limit theorem, such a projection (i.e., signal mixture) tends to appear to be "more Gaussian" than the components of the original vector [27], [28].

2) Anderson–Darling Test: The Anderson–Darling statistic test is a 1-D test which is based on the empirical cumulative distribution function and is one of the most powerful normality tests [29]. The following procedure shows how the Anderson–Darling test works:

1) Define two hypotheses: H_0: the data points around the dataset center are sampled from a Gaussian. H_1: the data points around the dataset center are not sampled from a Gaussian.
2) Normalize the data points so that their mean value becomes 0 and their variance equals 1. Let $x(i)$ be the ith ordered value. Let $F(x(i))$ be the $N(0,1)$ cumulative distribution function of the data points, with $z_i = F(x(i))$.
3) In our case, the mean value μ and standard value σ are estimated from the data, the corresponding test statistic is:

$$A_\star^2(Z) = \left(-\frac{1}{n}\sum_{i=1}^{n}(2i-1)*[\log z_i + \log(1-z_{n+1-i})] - n\right) * (1 + 4/n - 25/(n^2)). \quad (1)$$

4) Choose a confidence level α for the test; if $A_\star^2(Z)$ is a noncritical value for confidence level α, then accept H_0 and keep the original dataset center. Otherwise, reject H_0 and replace the original center with the new centers c_1, c_2; then assign the data to the two new centers.

D. Semi-supervised Learning

1) Cluster-Then-Label: With already homogeneously clustered data points on hand, we follow a cluster-then-label procedure within each cluster which then makes the whole procedure a semi-supervised classification [14].

To this end, the original cluster-then-label algorithm is modified to accelerate the image annotation. After the clusters are obtained, we label the training data within each cluster, which guarantees there is no cluster without training labels.

Algorithm 2. Modified cluster-then-label semi-supervised algorithm

Data: Labeled patches $(x_1, y_1), \ldots, (x_l, y_l)$, unlabeled patches $x_{l+1}, \ldots, x_{l+u}$, a clustering algorithm A, and a supervised learning algorithm L
Result: Labels for unlabeled patches $y_{l+1}, \ldots, y_{l+u}$
Use A to cluster $x_l, \ldots, x_l, x_{l+1}, \ldots, x_{l+u}$, obtain clusters $c_1, \ldots, c_n$.
for *each cluster* $c_i = c_1 : c_n$ **do**
  Let S_{c_i} be the labeled patches in the cluster c_i;
  Learn a supervised predictor from S_{c_i}: $f_{S_{c_i}} = L(S_{c_i})$;
  Apply $f_{S_{c_i}}$ to all unlabeled patches within the cluster c_i.
end

Algorithm 3. NBNN algorithm

Data: Feature descriptor types $f_1, \ldots, f_n$ of an image patch P.
Result: Class label $\tilde{C}_P$.
for *every* f_i **do**
  for *every defined* C_j **do**
    Compute the NN of f_i in class C_j: $NN_{C_j}(f_i)$.
  end
end
$\tilde{C}_P = \text{argmin}_C \sum_{i=1}^{n} \|f_i - NN_C(f_i)\|^2$.

Algorithm 2 describes the modified cluster-then-label algorithm.

2) Supervised Learning Within Clusters: For our studies, three algorithms are chosen and evaluated to perform supervised learning within clusters: a support vector machine (SVM) and a k nearest neighbor (KNN), as well as a Naive-Bayes Nearest Neighbor (NBNN) algorithm. We assume that the readers are already familiar with the SVM and KNN algorithms that rely on the image-to-image distance. Here, we explain the NBNN algorithm that obtains classifications from the perspective of image-to-class distances.

The NBNN image classifier is formalized in Algorithm 3, which is a highly accurate approximation of the optimal maximum a posteriori Naive-Bayes image classifier. A detailed theoretical derivation can be found in [30].

E. Comparative Similarity Measures

Since, the relationships among feature points in the feature space are measured by pair-wise distances, the definition of a distance metric hence plays an important role in exploring structures in feature space. As an extension of Euclidean space, the fractional distance and the L1 (i.e., Manhattan), L2 (i.e., Euclidean), and Lp (i.e., Minkowski) distance metrics are included in our k-means clustering algorithm and studied.

1) Fractional Distance Metric: The fractional distance metric is claimed to be able to tackle the "curse of dimensionality" problem [13], its distance parameter f lies within the range of $(0, 1)$

$$\mathrm{dist}_d^f(x, y) = \left[\sum_{i=1}^{d} |x^i - y^i|^f\right]^{1/f} \tag{2}$$

where d is the dimensionality of the feature space. It has been demonstrated in [13] that it performs better than the common Euclidean and Manhattan distance metrics (i.e., L2 and L1) in high-dimensional feature spaces, which is the normal case in image classification.

2) Minkowski Distance Metric: Similar to the fractional distance metric, the Minkowski distance [31] is a metric defined in Euclidean space which can be viewed as a generalization form of both the Euclidean and the Manhattan distances

$$\mathrm{dist}_d^p(x, y) = \left[\sum_{i=1}^{d} |x^i - y^i|^p\right]^{1/p} \tag{3}$$

where d is the dimensionality of the feature space and p is a parameter with $p \geq 1$. The Minkowski distance is typically used with p equal to 1 or 2 (i.e., Manhattan distance and Euclidean distance). When p reaches infinity, the so-called Chebyshev distance is obtained.

In our case, the range of distance parameters is set to [0.2, 2] with a step size of 0.2 and to [3, 13] with a step size of 1. In case of the original G-means algorithm, the range of distance parameters is modified to [0.8, 2] with a step size of 0.2 and to [3, 12] with a step size of 1, due to the long computational time.

F. K-Medoids Algorithm Implementation

With the alternative distance metrics in place of the traditional Euclidean distance metric, k-means turns into k-medoids. Thus, the cluster center is no longer the mean of the data points and we have to find the cluster center whose components are minimizers of the summed distances

$$\mathrm{dist}(C_k) = \sum_{i \in C_k} |y_i - c_k|^p. \tag{4}$$

In order to calculate the corresponding Minkowski center, a steepest descent algorithm is proposed in [32] and is claimed to converge much faster than a nature-inspired evolutionary method. Since, the Minkowski distance is very similar to a fractional distance (with $p >= 1$ or $0 < p < 1$), we extend the steepest descent algorithm to the entire real-valued parameter space (i.e., to $p > 0$).

G. Evaluation

Several tests are performed to evaluate the behavior of the proposed clustering method. We use quantitative measurements like internal criteria (e.g., the Dunn index) and external evaluations (i.e., precision/recall and F-score); we also make visual evaluations by analyzing feature space plots and the patches of selected cluster centroids.

1) Quantitative Evaluation: The clustering results can be evaluated in two ways:

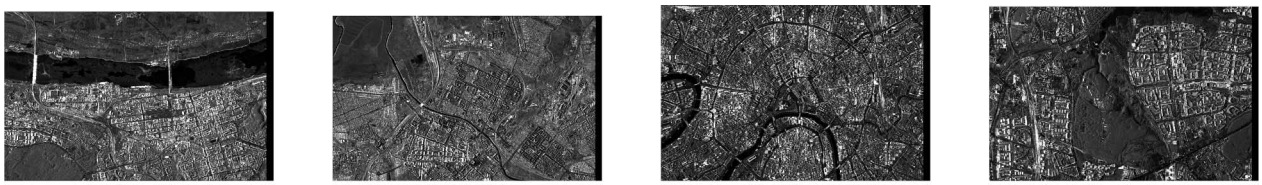
1) Internal evaluation: the clustering results are evaluated without reference data. The Dunn index that identifies dense and well-separated clusters is used in this paper. It is defined as the ratio between the minimal inter-cluster distance to the maximal intra-cluster distance [33]
$$D = \min_{1 \leq i \leq n} \min_{1 \leq j \leq n, i \neq j} \frac{d(i, j)}{\max_{1 \leq k \leq n} d'(k)}. \tag{5}$$
2) External evaluation: the clustering results are evaluated based on given reference data. In our case, the widely used confusion matrix, overall accuracy, and F-score are used to evaluate the classification accuracy. In addition, the overall accuracy and F-scores are also calculated for the original G-means algorithm. Equation (6) defines precision and recall; (7) shows the definition of F-score [34]
$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN} \tag{6}$$
$$F_\beta = \frac{((\beta)^2 + 1) \times P \times R}{(\beta)^2 \times P + R}. \tag{7}$$
Because each class generates a separate F-score value, a micro F-score (obtained by weighted averaging the class-specific F-scores) is used in this paper.
Finally, we present in this paper for each distance metric its individual computational time and the resulting number of clusters.

2) Visual Evaluation: Visual evaluation is used to get an intuitive idea about the clusters in feature space as it is very difficult to understand how features spread out in high-dimensional spaces [13]. Therefore, we have to apply a dimensionality reduction method to facilitate the understanding of the clusters. The following list describes our five visual evaluation methods.

1) Tree structure: The hierarchical clustering structure can be represented as a dichotomy tree, which explains the splitting of the clusters.
2) Feature space visualization: In order to visually evaluate the clustering results, the t-distributed stochastic neighbor embedding (t-SNE) algorithm can be used for dimensionality reduction; it retains the local data structure as well as the main global structure of the feature space [35]. For the reference data, the clusters are labeled using the available annotation classes; for the obtained clustering results, each cluster are shown in a different color.
3) Cluster centroid patches: For analyzing the compactness of clusters, the closest patch, the median distance patch, and the farthest patch from the cluster centroid are chosen to show the patch-feature relationships of the clusters.

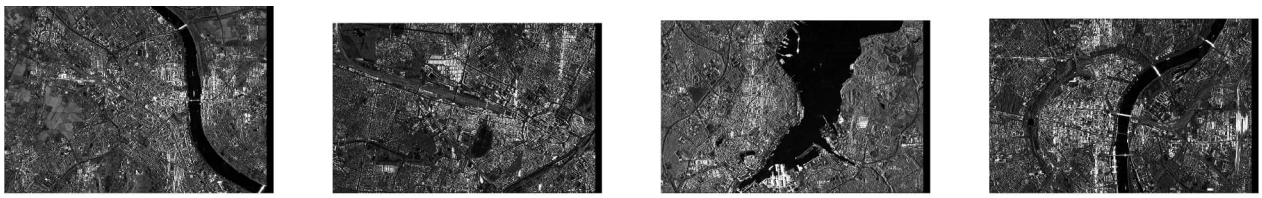
TABLE I
IMAGE DATA COLLECTION PARAMETERS

Dataset	Continent	Country	Resolution (m)	No. of scenes	No. of patches	No. of general classes	No. of detailed classes
Collection 02	Asia	Russia	2.9	7	7187	8	39
Collection 03	Europe	Germany, Switzerland	2.9	7	7176	8	41
Collection 17	North America	United States, Mexico	2.9	9	6975	8	30
Collection 27	Africa	South Africa, Zimbabwe, Nigeria, Togo	2.9	4	3536	8	22



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

Fig. 4. Examples of scenes from Russia [19], [23]. (a) Perm. (b) Tula. (c) Center of Moscow. (d) North Moscow.



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

Fig. 5. Examples of scenes from German-speaking countries. Here, the selected examples are from Germany [19], [23]. (a) Bonn. (b) Munich. (c) Kiel. (d) Cologne.

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

Fig. 6. Examples of scenes from North America [19], [23]. (a) Ciudad Juarez, Mexico. (b) Tucson, USA. (c) San Diego North, USA. (d) Sun Lakes, USA.

4) Cluster homogeneity: The cluster homogeneity is presented via pie charts, using the general and detailed level reference data, to provide the quantitative class percentages within each cluster.

III. RESULTS

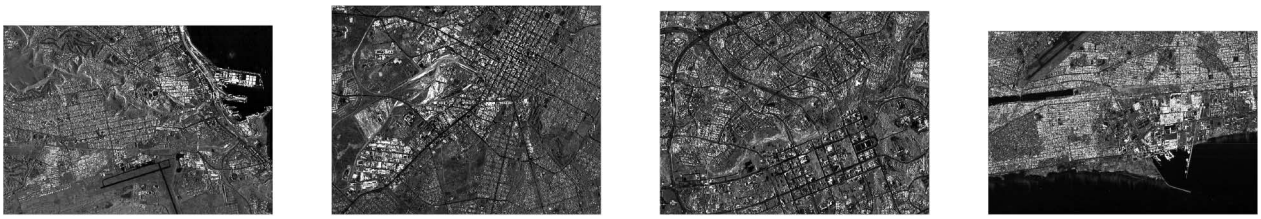
In this section, the selection of the image data collections, the subsampling of the pixels as well as the detailed parameter settings will be described. Then, the clustering results of the proposed methodology will be analyzed, including a quantitative evaluation to find the optimal distance metric, and also some visual evaluations which provide detailed insights into the data characteristics.

A. Image Data Selection and Subsampling

Prior to any data analysis, we had to select appropriate image data collections and to preprocess the data.

1) Data Selection: As typical examples, we selected four image collections from our database which cover different areas of the world from the North America, the Africa, and the Europe. Table I contains detailed information about the selected collections. Figs. 4–7 show some quick-look examples of images taken from the selected collections.

2) Data Preprocessing: Before we extracted feature descriptors, each image patch was subsampled from its original size of 160×160 to 80×80 pixels. This procedure decorrelated neighboring pixels, reduced the computational effort, and typically increased the recall by 1% [21].



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

Fig. 7. Examples of scenes from the Africa [19], [23]. (a) Port Elizabeth, South Africa. (b) Bulawayo, Zimbabwe. (c) Abuja, Nigeria. (d) Lome, Togo.

TABLE II
TABLE OF PARAMETER SETTINGS

Parameter	Value	Remark
Patch size	160×160 pixels	Depends on the imaging parameters of the TerraSAR-X data acquisitions (e.g., resolution and pixel spacing) [21].
Gabor filters	4 scales and 6 orientations	These parameters resulted from comparisons with other feature extraction methods and are also used in the MPEG-7 standard [24].
Confidence level in Anderson–Darling test	$\alpha = 0.0001$	Critical value = 1.8692.
Cluster size limit	$\geq$40 patches	The number 40 is chosen based on two reasons: 1) based on the Central Limit Theorem, to obtain a meaningful Gaussian distribution; 2) to obtain suitable cluster sizes (see Table I).
Supervision percentage	30 percent	Normally 30 percent is sufficient for experiments.

B. Parameter Settings

Table II lists the detailed parameter settings of our experiment.

C. Quantitative Evaluations

We started our performance evaluation experiment with image data collection 17 and our detailed level reference data. In this case, for urban areas, the given images contained high-density residential areas, medium density residential areas, low-density residential areas, skyscrapers, etc. [23]). In total, collection 17 contained 30 semantic classes.

From the experimental results, we presented not only the internal and external evaluations, but also the computational time and the resulting number of clusters. In order to explain the annotation results and to get a better understanding, we conducted a set of comparative tests to prove the effectiveness and limits of our proposed method. Finally, based on the quantitative results, the optimal distance metric is chosen for later experiments.

1) Internal Evaluation: Dunn's Index yielded 0.0047 for all distance metrics. Since, it was defined as the ratio of the minimal inter-cluster distance to the maximal intra-cluster distance, the constant result reflected that the obtained clustering structures are similar, even when applying different distance metrics.

2) External Evaluation: Fig. 8(a) and (c) depicts the overall classification accuracy of our detailed level reference data for each distance metric obtained with different supervised and semi-supervised classifiers (KNN, SVM, and NBNN), regarding the original and modified G-means algorithms.

In case of modified G-means algorithm, as shown by Fig. 8(a), we observe that:

1) When $p < 2$, all (supervised and semi-supervised) classification accuracies are decreasing versus distance; when $p > 2$, the accuracy does not change much.
2) When we compare the semi-supervised and supervised classifications, the supervised classifications are more accurate.

In case of the original G-means algorithm, as shown by Fig. 8(c), we observe that:

1) When $p < 2$, all (supervised and semi-supervised) classification accuracies are increasing versus distance; when $2 \leq p < 3$, all (supervised and semi-supervised) classification accuracies are decreasing versus distance; when $p \geq 3$, the accuracy does not change much.
2) When we compare the semi-supervised and supervised classifications, the semi-supervised classifications are more accurate.

Similarly, Fig. 8(b) and (d) shows the overall classification F-score of our detailed level reference data for each distance metric obtained with different supervised and semi-supervised classifiers (KNN, SVM, and NBNN), for the original and modified G-means algorithms, respectively. For the F-score plots, the remarks made to Fig. 8(a) and (c) applies, too.

Overall, the accuracy and F-score values of the original G-means algorithm are lower than the values of the modified G-means algorithm.

Fig. 9 shows the confusion matrix for a selected number of classes with a sufficient number of patch samples. The urban classes are often mixed up, which also reflects the "semantic gap" between manually annotated semantics and computer-based predictions.

3) Additional Evaluations: Besides the internal and external evaluations of the clustering results, we also considered the computational time and the number of generated clusters. Fig. 10 shows the computational time and the number of

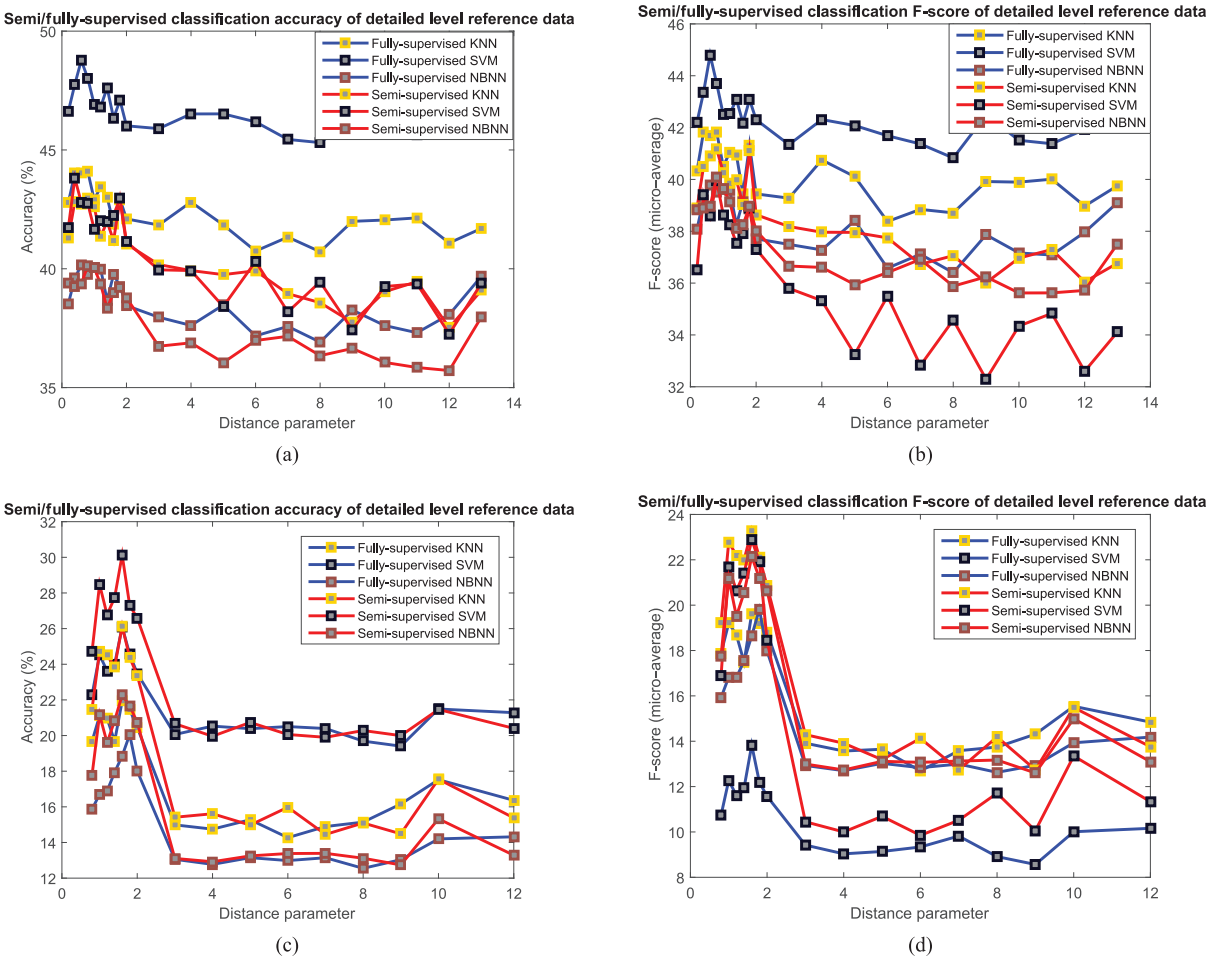


Fig. 8. Learning results of image data collection 17 and detailed level reference data. (a) Accuracy of detailed level reference data for the modified G-means algorithm. (b) Micro-average F-score of detailed level reference data for the modified G-means algorithm. (c) Accuracy of detailed-level reference data for the original G-means algorithm. (d) Micro-average F-score of detailed level reference data for the original G-means algorithm.

Actual \ predicted	Airport	Channel	Forest mixed	High-density residential area	Hill	Industrial area	Medium density residential area	Aerospace facilities	Mixed urban area	Ocean	Road	Skyscraper
Airport	66	6	2	0	17	1	0	6	0	22	3	0
Channel	13	23	4	0	12	11	5	13	4	0	17	0
Forest mixed	0	6	24	0	11	2	10	2	8	0	15	0
High-density residential area	0	0	0	165	0	16	29	0	3	0	3	15
Hill	5	9	7	0	84	0	0	3	1	4	15	0
Industrial area	0	4	0	41	0	96	63	12	14	0	34	9
Medium density residential area	0	2	5	35	0	40	703	0	92	0	38	8
Aerospace facilities	5	11	1	0	3	5	11	51	6	0	9	0
Mixed urban area	0	5	7	10	2	47	265	12	124	0	42	0
Ocean	47	2	0	0	8	0	0	1	0	207	0	0
Road	0	13	11	13	10	69	98	16	35	0	102	6
Skyscraper	0	0	0	31	0	34	19	0	3	0	10	102

Fig. 9. Confusion matrix of image data collection 17 with detailed level reference data for selected classes. Due to space limitations, we list only a number of selected classes.

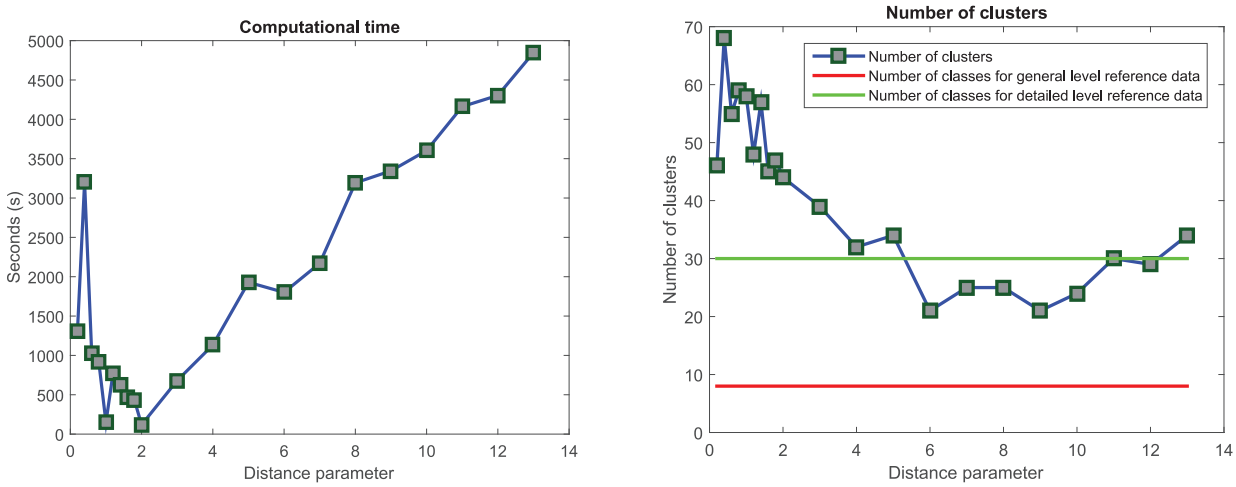
generated clusters for different distance metrics for the image data from collection 17.

Fig. 10(a) shows the computational time for all Lp distance results. We observe that:

1) For $p = 1$ and $p = 2$, the computational time is the lowest.
2) For $p < 1$ and $1 < p < 2$, the computational time is higher than for $p = 1$ and $p = 2$.
3) For $p > 2$, the computational time is increasing.

Fig. 10(b) presents the number of clusters for all Lp distance results. When we look at the classification accuracies of the generated clusters, we need to take into account that the general level reference data contain eight classes, detailed level reference data is 30. The results show that:

1) For $p < 6$, the number of clusters is fluctuating but the tendency is decreasing versus distance.



(a) (b)

Fig. 10. Evaluation of data collection 17. (a) Computational time. (b) Number of clusters. The number of annotated classes for the general level reference data is 8, the number of annotated classes for the detailed level reference data is 30.

2) For $6 < p < 12$, the number of clusters is lower than the attainable maximum (i.e., 30).

In general, with regard to detailed level reference data, semi-supervised classifiers deliver worse results than supervised classifiers. For the general level reference data, we have eight semantic classes with more than 120 patches per class; for the detailed level reference data, we have 30 semantic classes with around 15 patches per class. Hence, there are too few samples for each detailed class, so that the trained classifiers are too weak to make correct decisions.

Fig. 11(a) and (c) depicts the overall classification accuracy of the general level data for each distance metric, with KNN, SVM, and NBNN as classifiers, respect for the original and modified G-means algorithms. We notice for both algorithms:

1) For the semi-supervised classifiers, the highest classification accuracy is obtained by SVM, followed by KNN. The accuracy versus distance parameter is decreasing with higher distance parameter values.
2) For the supervised classifiers, we obtain a curve ordering similar to the semi-supervised curves. The tendency of the accuracy is fluctuating but stays relatively constant.
3) When we compare the semi-supervised classifiers with the supervised classifiers, the semi-supervised classifiers perform better.

Fig. 11(b) and (d) depicts the overall classification F-score of the general level reference data for each distance metric, using different supervised and semi-supervised classifiers (KNN, SVM, and NBNN), for the original and modified G-means algorithms. We observe that:

1) For the semi-supervised classifiers, we notice that the F-score decreases with higher values of Lp. The highest F-score value is obtained by KNN, followed by NBNN.
2) For the supervised classifiers, we obtain a curve ordering similar to the semi-supervised curves. The F-score value is fluctuating but stays relatively constant.
3) When we compare the semi-supervised classifiers with the supervised classifiers, the semi-supervised classifiers perform better.

For the accuracy and F-scores, the values of the modified G-means algorithm are higher than the original ones.

On the one hand, the results verified our guess that with more samples in each class, the cluster-then-label semi-supervised learning performs better than fully supervised learning. On the other hand, the need for a relatively large number of samples limits the applicability of the former method. To sum up, in terms of learning accuracy and F-score value, it is obvious that fractional distances give better performance. In terms of computational time, the classical Euclidean distance as well as the Manhattan distance cost much less time, while the fractional distances take around 100 times more time than the Euclidean and Manhattan norms. In terms of the number of clusters, Minkowski distances yield fewer clusters than the number of clusters for detailed level reference data, while fractional distances generate more and thus smaller clusters. As a consequence of the compromise between efficiency and performance, we chose $L = 1$ as our optimal distance parameter for subsequent analyses.

D. Visual Evaluations

All our visual results were generated with a distance metric of $L = 1$ for data collection 17.

1) Tree Structure: Fig. 12 shows the hierarchical tree structure with 4–10 layers by using our proposed method. A cluster that follows a Gaussian distribution and contains a sufficient number of patches is labeled as "1"; otherwise, it is labeled as "0." The homogeneities within the clusters (i.e., the zoomed area shown in Fig. 12) are analyzed in Section III-D4.

2) Feature Space Visualization: Using the t-SNE algorithm [35] which is claimed to preserve the complete local structure and some global structure of the data points, the original 48-dimensional feature space was reduced to three dimensions.

1) Figs. 13 and 14 show that due to the human interaction, the semantic annotations change the separating surfaces among the different classes, which then result in the spreading of a single class across the whole feature space.

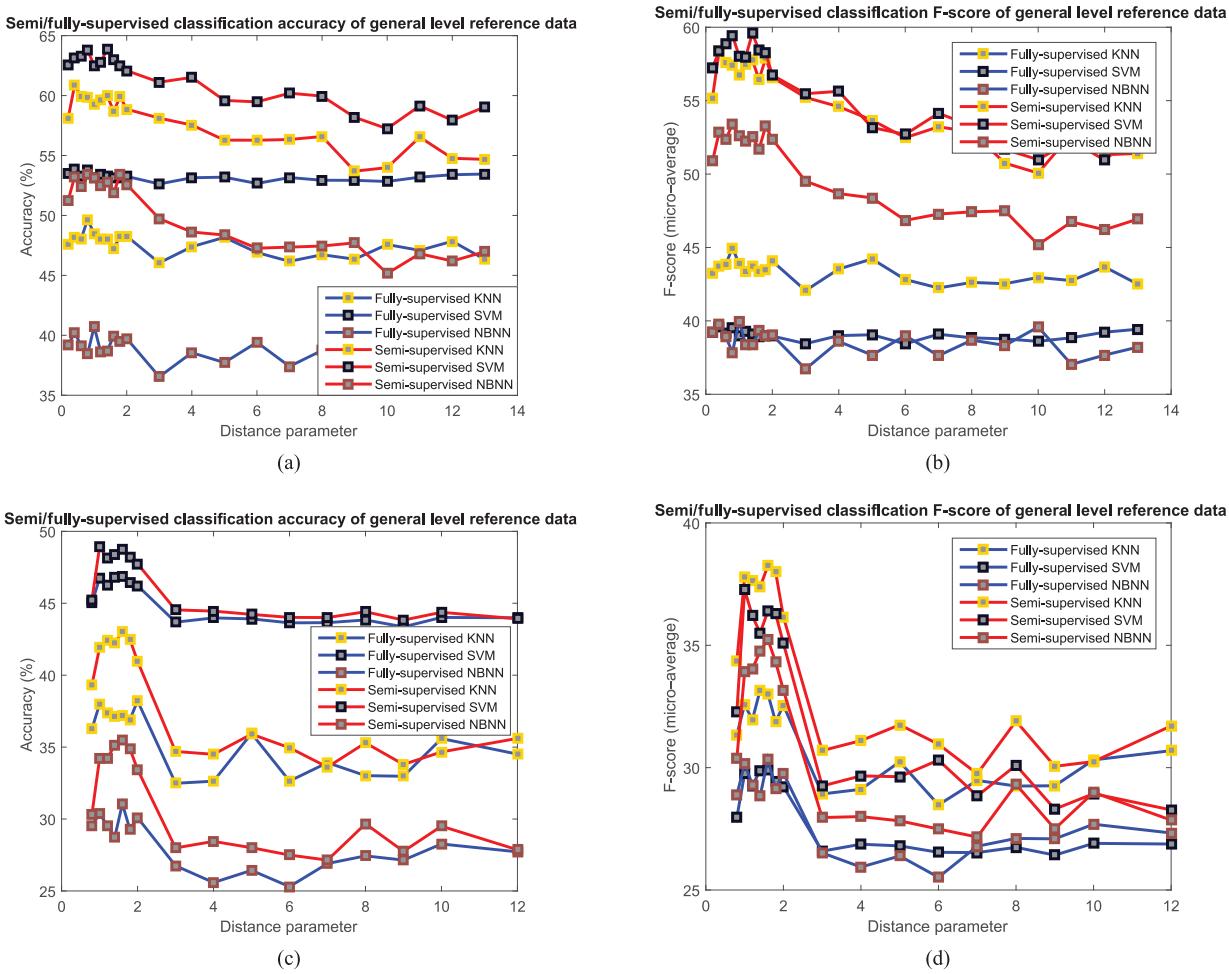


Fig. 11. Learning results of image data collection 17 and general level classification. (a) Accuracy of general level classifications for the modified G-means algorithm. (b) Micro-average F-score of general level classifications for the modified G-means algorithm. (c) Accuracy of general level classifications for the original G-means algorithm. (d) Micro-average F-score of general level classifications for the original G-means algorithm.

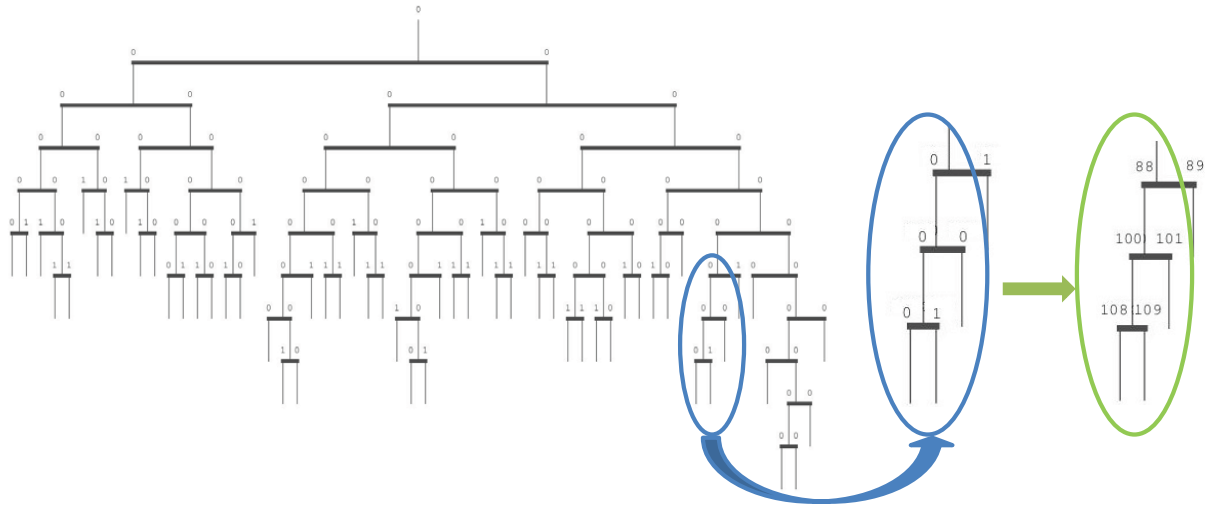


Fig. 12. Cluster tree structure of data collection 17. The blue ellipse shows some zoomed clusters, the green ellipse shows the corresponding cluster numbering.

2) Fig. 15 demonstrates the effectiveness of the cluster stopping criterion (i.e., the Gaussian hypothesis test), each cluster is visually compactly grouped.

3) Many classes are so spread out that without human interaction or supervision, it is impossible to rely on unsupervised learning methods to separate classes.

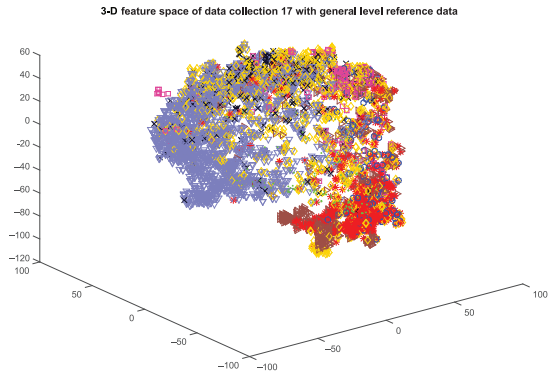


Fig. 13. 3-D feature space of the data collection 17 with general level reference data. The different colors stand for classes with different semantic annotations. The axis scaling refers to projected t-SNE results [35].

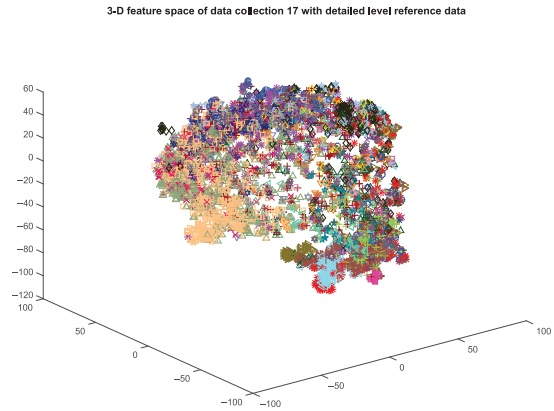


Fig. 14. 3-D feature space of the data collection 17 with detailed level reference data. The different colors stand for classes with different semantic annotations. The axis scaling refers to projected t-SNE results [35].

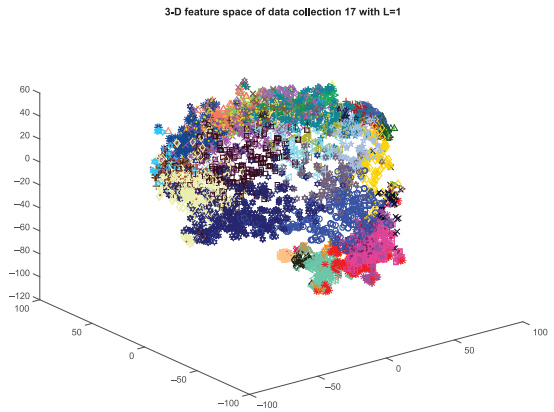


Fig. 15. 3-D feature space of the data collection 17 with a distance parameter of 1. The different colors stand for different obtained clusters. The axis scaling refers to projected t-SNE results [35].

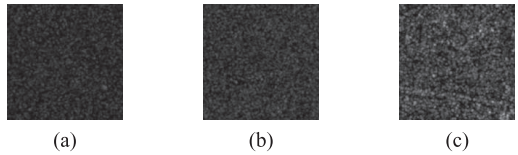


Fig. 16. Patch examples of the most compact cluster (collection 17). (a) Patch closest to the cluster center. (b) Mid-distance patch from the cluster center. (c) Patch farthest from the cluster center.

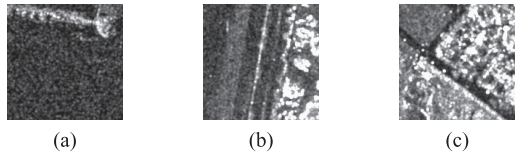


Fig. 17. Patch examples of the mid-compact cluster (collection 17). (a) Patch closest to the cluster center. (b) Mid-distance patch from the cluster center. (c) Patch farthest from the cluster center.

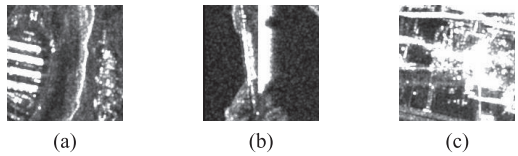


Fig. 18. Patch examples of the most spread out cluster (collection 17). (a) Patch closest to the cluster center. (b) Mid-distance patch from the cluster center. (c) Patch farthest from the cluster center.

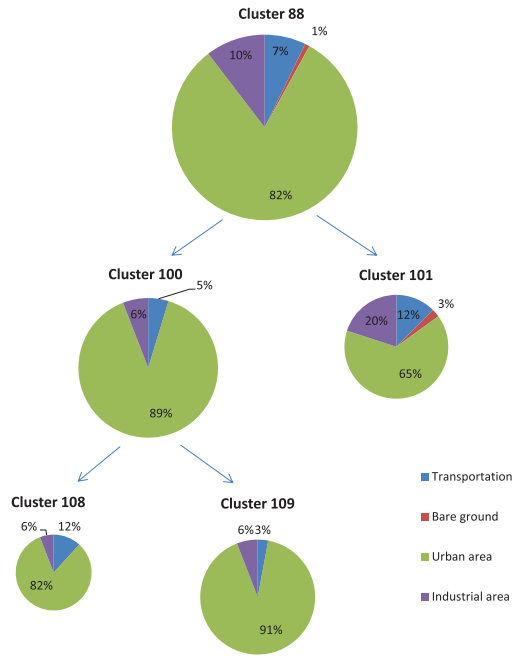


Fig. 19. Cluster splitting for the data collection 17 with the general level reference data.

3) Cluster Centroid Patches: We chose three scales of cluster distances: most compact, mid-compact and spread out. For each cluster, the distance-based feature-patch correspondence was illustrated by three representative patches: the closest patch, the mid-distance patch, and the farthest patch from the cluster center. Figs. 16–18 illustrate how the distance influences the representative patches of the clusters.

1) Fig. 16 shows the patches with very low intensity which correspond to water body classes.

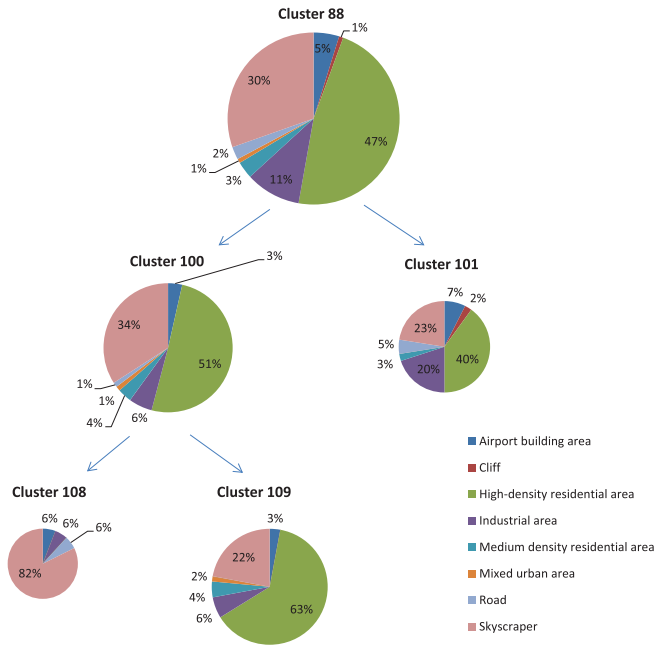


Fig. 20. Cluster splitting for the data collection 17 with the detailed level reference data.

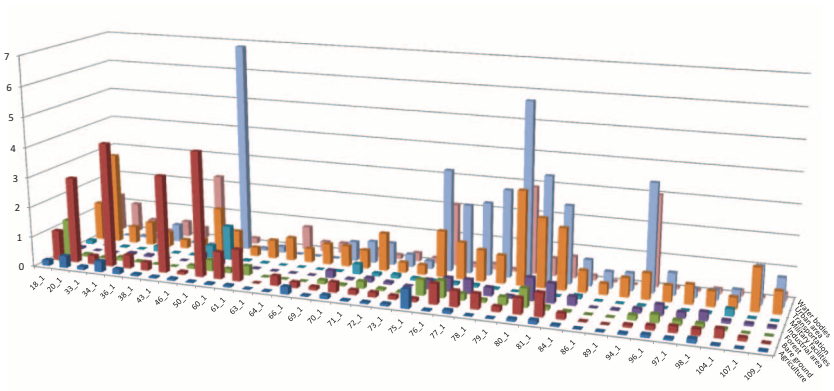


Fig. 21. Class-cluster distribution of data collection 17 with general level reference data. *x*-axis represents the last layer Gaussian distributed cluster number; *y*-axis shows the general level reference class labels; *z*-axis represents the percentage of the number of patches versus the total number of patches.

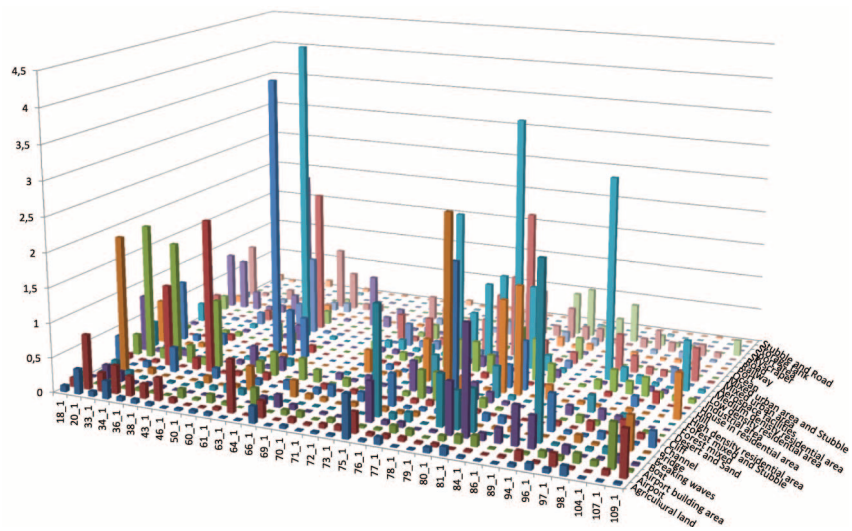


Fig. 22. Class-cluster distribution of data collection 17 with detailed level reference data. *x*-axis represents the last layer Gaussian distributed cluster number; *y*-axis shows the detailed level reference class labels; *z*-axis represents the percentage of the number of patches versus the total number of patches.

2) Fig. 17 shows the patches with mid-intensity which correspond to transportation or agriculture classes.
3) Fig. 18 shows the patches with very high intensity which correspond to urban areas.
4) Figs. 16–18 demonstrate that high-intensity urban area classes tend to yield spread out clusters; in contrast, water bodies and agriculture classes with low intensity tend to result in compact clusters.

4) Cluster Homogeneity:

1) Figs. 19 and 20 show the layer-wise splitting of clusters for general and detailed level reference data, which corresponds to the zoomed clusters in Fig. 12. The size of the pie-diagrams reflect the number of patches within each cluster. Clusters 88, 100, and 109 represent the main route of cluster splitting, where urban areas are the dominant class in the general level reference data, and high-density residential areas are the dominant class in the detailed level reference data.
2) Figs. 21 and 22 show the overall class-cluster distribution for the general and detailed level reference data, respectively. For the general level reference data, most of the clusters contain one or two dominant classes; for the detailed level reference data, due to the increased number of classes, the distributions tend to be more disordered than for the general level reference data. However, there are a number of clusters with highly dominant classes, which appear to be very homogeneous.

IV. Conclusion

In this paper, we have proposed a processing and analyzing procedure for large-scale SAR image annotation, which is illustrated in Fig. 1. For the purpose of testing and evaluation, four semantically annotated data collections with two-level reference data was prepared. We used them for the analysis of two methods proposed in the introduction, a hierarchical cluster splitting method and various distance metrics (fractional and Minkowski distances) in order to explore the information contained in the feature space. We compared the semi-supervised results and the annotated reference data and analyzed the relations between the clustering results and the general level reference as well as the relations between the semi-supervised classification results and the detailed level reference data. Based on quantitative and visual evaluations of the experimental results, we compared relations among the clustering results, the semi-supervised classification results, and the general and detailed level reference data. It turned out that our proposed method is able to obtain reliable results for the general level reference data; however, due to the too many detailed subclasses and their few instances, the proposed method generates inferior results for the detailed level reference data. Similar results were obtained for all data collections.

A. Clustering

There were two main issues when we analyzed unsupervised clustering: the distance metrics (i.e., fractional distance, Minkowski distance) and the termination criterion for the cluster splitting (i.e., the Gaussian hypothesis test).

Regarding the overall classification accuracy and F-score, fractional distances outperformed Minkowski distances. The stop criterion worked well; when we visualized the feature space, the clusters were grouped compactly; during cluster splitting, the clusters tended to become homogeneous with one or two dominant classes. This was seen in the pie diagrams. Moreover, by observing the quick-look patches of the clusters it turned out that most of the clusters were very homogeneous, although they may have different semantic labels.

As demonstrated by Figs. 8 and 11, the modified G-means algorithm performed better than the original G-means algorithm.

B. Classifiers

For the classification accuracy and F-score value, we investigated the performances of different supervised learning methods that was used within the clusters (i.e., SVM, KNN, and NBNN).

As for the overall classification accuracy, SVM always achieved the best results both for general and detailed level reference data. In the case of the F-score value, the nearest neighbor methods (i.e., KNN and NBNN) performed better than SVM. The reason behind it was that SVM obtains better classification accuracies for some dominant classes, but lower classification accuracies for the other minority classes.

When the number of classes within a cluster was increasing (e.g., semi-supervised NBNN compared to semi-supervised KNN, for F-score values of detailed level reference data), NBNN performed better than KNN. This indicated that NBNN tended to provide good results when we had a limited number of samples in each class.

C. Semi-supervised Learning and Manually Annotated Reference Data

As mentioned in [4], unsupervised clustering made weaker demands on the quality of the manual annotations which largely reduced the human effort. However, it did not explicitly treat semantics as image classes; therefore, it was not guaranteed that the semantic annotations were optimal in a recognition or retrieval sense.

In order to bridge this "semantic gap," the relationships between the unsupervised clusters and the general and detailed level reference data had to be discussed:

When we looked at the relations between clustering results and general level reference data, and the relations between clustering results and detailed level reference data, usually a cluster comprised 5–6 classes, with one or two of them being dominant. Of course, the correspondence was better for the general level reference data due to their lower number of classes. With a good classifier, each patch was labeled correctly with a probability of 0.6 for the general level reference data, and with a probability of 0.4 for the detailed level reference data.

D. Semiannotation/Labeling

In the end, we provide a general framework which can be used as a reference by other practitioners who are also

interested in large-scale SAR image dataset annotations. We recommend the following procedure:

1) Tile the dataset into image patches and extract patch features.
2) Use clustering methods to group feature data into clusters based on distance similarity. It is better to use fractional distance or Manhattan distance rather than the traditional Euclidean distance.
3) Within each cluster, label some general classes manually; then use a supervised learning method to label the remaining unlabeled patches.

E. Future work

Although the annotation accuracy is increased by the proposed method, there is still much space left for future more efficient annotation of large-scale SAR image datasets.

Based on the results of this publication, more work shall be invested in feature extraction and learning methods. For the extracted multiorientation and multiscale Gabor coefficients, more statistics can be calculated: mean, median, mode, variance, standard deviation, interquartile range, skewness, and kurtosis. Furthermore, other types of good texture extractors are also worth to try. Regarding the learning part, in order to bridge the semantic gap, we plan to add a certain degree of human interaction or supervision, e.g., active learning.

ACKNOWLEDGMENT

The image data being used in this study were provided by the TerraSAR-X Science Service System (Proposal MTH 1118). The authors would like to thank G. Schwarz for many helpful hints and the reviewers for their valuable and insightful comments.

REFERENCES

[1] A. Barriuso and A. Torralba, "Notes on image annotation," *Comput. Res. Repository (CoRR)*, 2012.
[2] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *Int. J. Comput. Vis.*, vol. 77, pp. 157–173, 2008.
[3] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "SUN database: Large-scale scene recognition from abbey to zoo," in *Proc. IEEE Comput. Vis. Pattern Recog. (CVPR)*, 2010, pp. 3485–3492.
[4] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 394–410, Mar. 2007.
[5] M. Lienou, H. Maitre, and M. Datcu, "Semantic annotation of satellite images using latent Dirichlet allocation," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 28–32, Jan. 2010.
[6] J. H. Su, C. L. Chou, C. Y. Lin, and V. S. Tseng, "Effective semantic annotation by image-to-concept distribution model," *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 530–538, Jun. 2011.
[7] P. Blanchart and M. Datcu, "A semi-supervised algorithm for auto-annotation and unknown structures discovery in satellite image databases," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 3, no. 4, pp. 698–717, Dec. 2010.
[8] P. Blanchart, M. Ferecatu, S. Cui, and M. Datcu, "Pattern retrieval in large image databases using multiscale coarse-to-fine cascaded active learning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 4, pp. 1127–1141, Apr. 2014.
[9] W. Luo, H. Li, and G. Liu, "Automatic annotation of multispectral satellite images using author–topic model," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 4, pp. 634–638, Jul. 2012.
[10] Z. Zhang, M. Y. Yang, M. Zhou, and X.-Z. Zeng, "Simultaneous remote sensing image classification and annotation based on the spatial coherent topic model," in *Proc. Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2010, pp. 1698–1701.
[11] K. Chen, P. Jian, Z. Zhou, J. Guo, and D. Zhang, "Semantic annotation of high-resolution remote sensing images via Gaussian process multi-instance multilabel learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 6, pp. 1285–1289, Nov. 2013.
[12] A. Ulges, M. Worring, and T. Breuel, "Learning visual contexts for image annotation from Flickr groups," *IEEE Trans. Multimedia*, vol. 13, no. 1, pp. 330–341, Apr. 2011.
[13] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *Proc. Int. Conf. Database Theory (ICDT'01)*, 2001, pp. 420–434.
[14] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synth. Lect. Artif. Intell. Mach. Learn.*, vol. 3, pp. 31–33, 2009.
[15] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[16] M. Datcu and G. Schwarz, "Image information mining methods for exploring and understanding high resolution images," in *Proc. Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2010, pp. 33–35.
[17] A. W. Smeulders, M. Worring, S. Simone, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.
[18] R. Bahmanyar and M. Datcu, "Measuring the semantic gap based on a communication channel model," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2013, pp. 4377–4381.
[19] DLR (2013). *TerraSAR-X Level 1b Product Format Specification* [Online]. Available: http://www2.geo-airbusds.com/files/pmedia/public/r460_030201_level-1b-product-format-specification_1.3.pdf
[20] S. Newsam, L. Wang, S. Bhagavathy, and B. Manjunath, "Using texture to annotate remote sensed datasets," in *Proc. 3rd Int. Symp. Image Signal Process. Anal. (ISPA'03)*, 2003, vol. 1, pp. 72–77.
[21] C. O. Dumitru and M. Datcu, "Information content of very high resolution SAR images: Study of feature extraction and imaging parameters," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 8, pp. 4591–4610, Aug. 2013.
[22] S. Cui, C. Dumitru, and M. Datcu, "Semantic annotation in Earth observation based on active learning," *Int. J. Image Data Fusion*, vol. 5, pp. 152–174, 2014.
[23] C. Dumitru, S. Cui, and M. Datcu, "Information content of very high resolution SAR images: Semantics, geospatial context, and ontologies," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 4, pp. 1635–1650, Nov. 2014.
[24] *MPEG-7* [Online]. Available: http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm.
[25] H. Greg and E. Charles, "Learning the K in K-means," in *Proc. 7th Annu. Conf. Neural Inf. Process. Syst.*, 2003, pp. 281–288.
[26] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discret. Algorithms*, 2007, pp. 1027–1035.
[27] J. Kruskal, "Toward a practical method which helps uncover the structure of a set of observations by finding the line transformation which optimizes a new "index of condensation"," in *Stat. Comput.*, R. C. Milton and J. A. Nelder Eds., New York: Academic Press, pp. 427–440, 1969.
[28] J. V. Stone, *Independent Component Analysis: A Tutorial Introduction*. Cambridge, MA, London, England: MIT Press, 2004.
[29] T. W. Anderson and D. A. Darling, "Asymptotic theory of certain "goodness of fit" criteria based on stochastic processes," *Ann. Math. Stat.*, vol. 23, pp. 193–212, 1952.
[30] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *Proc. 26th IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2008, pp. 1–8.
[31] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, pp. 1–27, 1964.
[32] R. C. Amorim and B. Mirkin, "Minkowski metric, feature weighting and anomalous cluster initializing in K-means clustering," *Pattern Recognit.*, vol. 45, pp. 1061–1075, 2012.
[33] J. Dunn, "A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters," *J. Cybern.*, vol. 3, pp. 32–57, 1973.
[34] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.*, vol. 45, pp. 427–437, 2009.
[35] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

Wei Yao received the B.Eng. degree in biomedical engineering and the M.Sc. degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology (HUST), Wuhan, China, in 2008 and 2011, respectively. She is currently pursuing the Ph.D. degree in electrical engineering at the University of Siegen, Siegen, Germany.

She has been a Guest Researcher with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, Germany, since August 2014. Her research interests include image feature extraction, SAR image statistical modeling, and machine learning.

Corneliu Octavian Dumitru received the B.S. and M.S. degrees in applied electronics and the Ph.D. degree in engineering from the Politehnica University Bucharest, Bucharest, Romania, in 2001, 2002, and 2006, respectively, and the second Ph.D. degree in telecommunications from Pierre and Marie Curie University, Paris, France, in 2010.

He had a teaching activity as a Lecturer with the Politehnica University, delivering lectures and seminaries and supervising laboratory works in the fields of information and estimation theory, communication theory, and signal processing. From 2005 to 2006 and in 2008, he was a Coordinator for two national grants delivered by the Romanian Ministry of Education and Research. Since 2010, he has been a Scientist with the Remote Sensing Technology Institute (IMF), Earth Observation Center (EOC), German Aerospace Center (DLR), Oberpfaffenhofen, Germany. Recently, he is involved in several projects in the frame of the European Satellite Agency (ESA) and European FP7/H2020 Programs for information extraction, taxonomies, and data mining using remote sensing imagery. His research interests include stochastic process information, model-based sequence recognition and understanding, basics of man–machine communication, information managing, and retrieval in extended databases.

Otmar Loffeld (M'05–SM'06) received the Diploma degree in electrical engineering from the Technical University of Aachen, Aachen, Germany, in 1982, and the Eng.Dr. degree and the "Habilitation" in digital signal processing and estimation theory from the University of Siegen, Siegen, Germany, in 1986 and 1989, respectively.

In 1991, he became a Professor of Digital Signal Processing and Estimation Theory with the University of Siegen. He lectures on general communication theory, digital signal processing, stochastic models and estimation theory and synthetic aperture radar, and has authored two textbooks on estimation theory. In 1995, he joined the Center for Sensorsystems (ZESS), University of Siegen, and became the Chair in 2005. In 1999, he became a Principal Investigator (PI) on Baseline Estimation for the X-Band part of the Shuttle Radar Topography Mission (SRTM) to which ZESS contributed to DLR's baseline calibration algorithms. He is a PI for interferometric techniques in the German TerraSAR-X mission, and a PI for a bistatic spaceborne airborne experiment, where TerraSAR-X serves as the bistatic illuminator while FGAN's PAMIR system mounted on a Transall airplane is used as a bistatic receiver. In 2002, he founded the International Postgraduate Program (IPP) "Multi Sensorics," and in 2008 established the "NRW Research School on Multi Modal Sensor Systems for Environmental Exploration and Safety (MOSES)" with the University of Siegen. His research interests include multisensor data fusion, Kalman filtering for data fusion, optimal filtering and process identification, SAR processing and simulation, SAR-interferometry, phase unwrapping, baseline estimation and, recently, bistatic SAR processing and compressive sensing.

Dr. Loffeld is a member of the ITG/VDE and a Senior Member of the GRSS.

Mihai Datcu (SM'04–F'13) received the M.S. and Ph.D. degrees in electronics and telecommunications from the University Politechnica Bucharest (UPB), Bucharest, Romania, in 1978 and 1986, respectively, and the Habilitation à diriger des recherches in computer science from the University Louis Pasteur, Strasbourg, France, in 1999.

Since 1981, he has been a Professor with the Department of Applied Electronics and Information Engineering, Faculty of Electronics, Telecommunications and Information Technology (ETTI), UPB, working in image processing and electronic speckle interferometry. Since 1993, he has been a Scientist with German Aerospace Center (DLR), Oberpfaffenhofen, Germany. He is developing algorithms for model-based information retrieval from high complexity signals and methods for scene understanding from very high resolution synthetic aperture radar (SAR) and interferometric SAR data. He is engaged in research related to information theoretical aspects and semantic representations in advanced communication systems. Currently, he is a Senior Scientist and an Image Analysis Research Group Leader with the Remote Sensing Technology Institute (IMF), DLR. Since 2011, he has also led the Immersive Visual Information Mining research lab at Munich Aerospace and he is director of the Research Center for Spatial Information at UPB. He has held Visiting Professor appointments at the University of Oviedo, Oviedo, Spain, the University Louis Pasteur, and the International Space University, both in Strasbourg, France; the University of Siegen, Siegen, Germany; the University of Innsbruck, Innsbruck, Austria; the University of Alcala, Madrid, Spain; the University Tor Vergata, Rome, Italy; the Universidad Pontificia de Salamanca, Madrid, Spain; the University of Camerino, Camerino, Italy; and the Swiss Center for Scientific Computing (CSCS), Manno, Switzerland, From 1992 to 2002, he had a longer Invited Professor assignment with Swiss Federal Institute of Technology, ETH Zurich, Switzerland. Since 2001, he had initiated and led the Competence Centre on Information Extraction and Image Understanding for Earth Observation, ParisTech, Paris Institute of Technology, Telecom Paris, Paris, France, a collaboration of DLR with the French Space Agency (CNES). He has been a Professor Holder of the DLR-CNES Chair with ParisTech, Paris Institute of Technology, Telecom Paris. He initiated the European frame of projects for Image Information Mining (IIM) and is involved in research programs for information extraction, data mining and knowledge discovery and data understanding with the European Space Agency (ESA), NASA, and in a variety of national and European projects. He has authored more than 450 scientific publications, among them about 80 journal papers, and a book on number theory. He and his team have developed and are currently developing the operational IIM processor in the Payload Ground Segment systems for the German missions TerraSAR-X, TanDEM-X, and the ESA Sentinel 1 and 2. His research interests include Bayesian inference, information and complexity theory, stochastic processes, model-based scene understanding, image information mining, for applications in information retrieval and understanding of high resolution SAR, and optical observations.

Dr. Datcu is a member of the ESA Big Data from Space Working Group. He has served as a Co-Organizer of international conferences and workshops, and as a Guest Editor of a special issue on IIM of the IEEE and other journals. He is the Representative of Romania in the ESA Industrial Policy Committee (IPC) and Earth Observation Programme Board (EO-PB). He was the recipient of the Best Paper Award in 2006, the IEEE Geoscience and Remote Sensing Society Prize in 2008, the National Order of Merit with the rank of Knight, for outstanding international research results, awarded by the President of Romania, and in 1987, the Romanian Academy Prize Traian Vuia for the development of SAADI image analysis system and activity in image processing.