

Accelerated Probabilistic Learning Concept for Mining Heterogeneous Earth Observation Images

Kevin Alonso and Mihai Datcu, *Fellow, IEEE*

Abstract—We present an accelerated probabilistic learning concept and its prototype implementation for mining heterogeneous Earth observation images, e.g., multispectral images, synthetic aperture radar (SAR) images, image time series, or geographical information systems (GIS) maps. The system prototype combines, at pixel level, the unsupervised clustering results of different features, extracted from heterogeneous satellite images and geographical information resources, with user-defined semantic annotations in order to calculate the posterior probabilities that allow the final probabilistic searches. The system is able to learn different semantic labels based on a newly developed Bayesian networks algorithm and allows different probabilistic retrieval methods of all semantically related images with only a few user interactions. The new algorithm reduces the computational cost, overperforming existing conventional systems, under certain conditions, by several orders of magnitude. The achieved speed-up allows the introduction of new feature models improving the learning capabilities of knowledge-driven image information mining systems and opening them to Big Data environments.

Index Terms—Active learning (AL), bag-of-words (BoW), Bayesian networks, Big Data, data fusion, geographical information systems (GIS), image mining.

I. INTRODUCTION

THE AMOUNT of Earth observation (EO) data is constantly increasing. This is due to the growing number of EO missions in space and their instrument characteristics that have evolved continuously. In addition, a lot of the currently available EO instruments offer very high spatial resolution data. As for imaging sensors, the well-known panchromatic and multispectral images have been complemented by hyperspectral images and a wide range of synthetic aperture radar (SAR) images generated with different techniques such as polarimetric SAR (PolSAR) or interferometric SAR (InSAR). The nature of the imagery is not the only factor that should be taken into account. EO products also comprise of metadata providing useful additional information such as satellite orbit state vectors, geographical coordinates, and data acquisition times. Further, we can also consider the available information coming from third party systems not directly related to satellite EO products. Prominent examples of a third party system are the widely spread geographical information systems (GIS) [1], [2], which store map information that can be used for different purposes

during EO image analysis. In a Big Data heterogeneous scenario, the challenge for the remote sensing community is to exploit existing EO archives more efficiently. By means of data mining, we can develop new systems, tools, techniques, algorithms and concepts; we obtain content-based image retrieval (CBIR) systems capable of quick extraction of valuable information for a better understanding of EO image content. CBIR as defined in [3] is the union of technologies which aim to help in the management of image and video digital archives by means of their visual content. Thus, anything ranging from a simple image similarity function to more complex image annotation engines can be seen as part of CBIR. CBIR search engines for EO applications can be divided into three main categories according to their functionality: 1) query by example (QbE); 2) relevance feedback (RF); and 3) active learning (AL). A QbE engine uses an image for a query and, based on its features, returns a ranking of the database content. An example of this type of engine is GeoIRIS [4]. In contrast, RF systems allow the user to refine a given query by iteratively specifying a set of relevant and a set of nonrelevant images. A multimedia application of this type of search engines is IKONA [5]. Finally, an AL engine requires interactive user action in order to refine the query parameters to obtain the desired results. The knowledge-driven information mining (KIM) system presented in [6] is an example of an AL system. KIM requires interaction of the users who provide semantic interpretation of the image content, which is internally linked to a hierarchical Bayesian network. The user can query the database for relevant images and obtain a probabilistic ranking of the entire image archive as an intuitive information representation. During the last years, several systems have been developed with different technology approaches in order to handle EO image heterogeneity and its characteristics. Different QbE paradigms have been proposed to retrieve multispectral images like the region-based image retrieval system developed at Oak Ridge National Laboratory in [7] and the multisensor evolution analysis (MEA) [8]. We can also find in the literature systems like the previously mentioned GeoIRIS and the intelligent interactive knowledge retrieval (I³KR) [9] that try to retrieve images by means of semantics. There are also systems for the retrieval and analysis of SAR and corresponding images time series. Selected examples of these systems are KIM [6], image information mining in time series (IIM-TS) [10], and the PicSOM system based on self-organizing maps (SOM) [11]. Recently, different research projects like EOLib [12] or TELEIOS [13] have introduced the use of EO image metadata and linked data as query parameters in order to improve the results. Linked data can be seen as a collection of best practices for publishing semantically structured

Manuscript received September 30, 2014; revised April 17, 2015; accepted May 11, 2015.

The authors are with German Aerospace Center, Wessling 82234, Germany (e-mail: kevin.alonso@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.2015.2435491

and interrelated datasets on the Web [14]. A review of current EO IIM systems can be found in [15].

This paper aims to extend the work presented in [16] where GIS map information was used in the AL stage of the image mining system. We present in a more extensive way the accelerated probabilistic learning concept for mining heterogeneous EO Big Data. We also define the system prototype implementation which is based on the principles proposed in the KIM system such as feature fusion and Bayesian networks, used for learning and probabilistic retrieval. For the learning stage, we present a modification of the Bayesian algorithm which achieves a speed-up of several orders of magnitude, making it suitable for a Big Data environment. The acceleration also allows the fusion of additional feature models (i.e., the BoW of every clustered set of features) during the learning process. Moreover, the presented approach makes use of EO data heterogeneity not only by fusing features extracted from different sensors, but also by linking GIS repositories. Our use of linked data goes further than the approaches proposed in [12] and [13] since it is included in the AL stage to improve the user specific semantics, and not just as a simple parameter to refine the queries. Besides, in the probabilistic retrieval stage, we introduce a new retrieval method based on the calculation of the similarity between probabilities that has been proved to offer good results with less learning interactions.

This paper is organized as follows. Section II describes the main aspects behind a classical KIM system. Section III introduces the heterogeneous data mining (HDM) concept followed by the sections presenting the elements that is composed of the HDM system. Specifically, Section IV describes the feature extraction processes and Section V the feature clustering and the generation of the bag-of-words (BoW). Continuing with the HDM modules, Section VI introduces the machine learning methods implemented. Section VII explains the available retrieval methods followed by the introduction of the user interface (UI) in Section VIII, and the description of the user interactions in Section IX. Section X demonstrates the system capabilities for different application scenarios, while Section XI presents the most important system performance parameters. Finally, Section XII contains general conclusion.

II. KIM

A classical KIM system, like the one presented in [17], represents the managed information via a multilevel hierarchical model, as shown in Fig. 1. Its initial level is formed by the data sources D , for instance, different EO image data. The second level of the hierarchical model contains the first processed information θ extracted from the EO products via statistical or mathematical analysis. For pixel-wise analysis, and being usually θ high dimensional, this stage generally increases the amount of managed data by various orders of magnitude. The third information level is composed of clustered θ features. The resulting cluster identifiers can be interpreted as words ω . In this stage, the data quantity is reduced from the highly dimensional features to an easily manageable single layer of words. At this point, it is possible to represent D by a normalized histogram of word occurrences. Finally, in the last level of the hierarchical

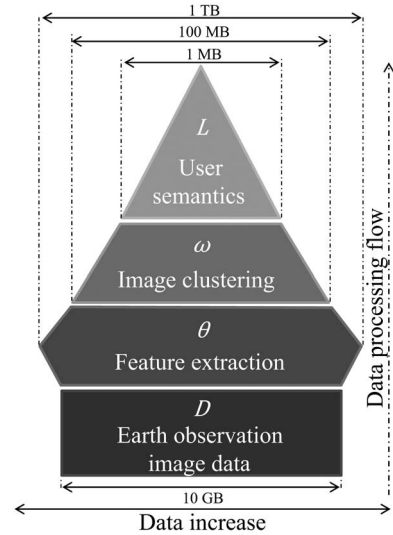


Fig. 1. System hierarchical levels. The initial level is formed by the data sources D . The second level refers to the information extracted via mathematical analysis θ . Third information level represents the clustering results ω of the features θ extracted in the previous level. The last level consists of the user's specific concept L and its relationship with the existent words.

model, the user attaches semantic labels L to the existing words.

In KIM, the user introduces semantic interpretation of a selected image (sub)scene via AL, by giving positive and negative examples, which are interactively linked with a hierarchical Bayesian network [18] (not shown in Fig. 1) to a content-index formed by a combination of different ω_i , where i is an index for the available words. Using the different words ω_i for all extracted features θ , one can identify an image uniquely by means of the probabilities $p(\omega_i|D)$, which express the occurrences of the words within the given image. These words will result in forming a BoW [19]. The Bayesian network allows the user to interactively define a link between a specific semantic label L with the existing words ω_i known as the stochastic link $p(\omega_i|L)$. Once the stochastic link is defined, it is possible to calculate the posterior probability $p(L|D)$, which is used to query the database for relevant images, obtaining a probabilistic ranking of the entire image archive as a semantic information representation. A brief description of the theoretical aspects is given in Section VI.

III. HDM CONCEPT

The HDM conceptual design, as any Bayesian inference system, is composed of two main stages: 1) a data-driven initial stage; and 2) the final user-driven stage. In Fig. 2, the conceptual stages, their independent modules and connections are shown. The initial modules of the data-driven stage represent the different heterogeneous EO databases or GIS repositories. From these online linked repositories, HDM gets the data to be analyzed. The analysis stage itself is performed offline. In the first analysis step, different types of features can be extracted at pixel level. In addition, the information available in the GIS repository in form of a vector map is rasterized to the EO image resolution. All the extracted features are clustered automatically

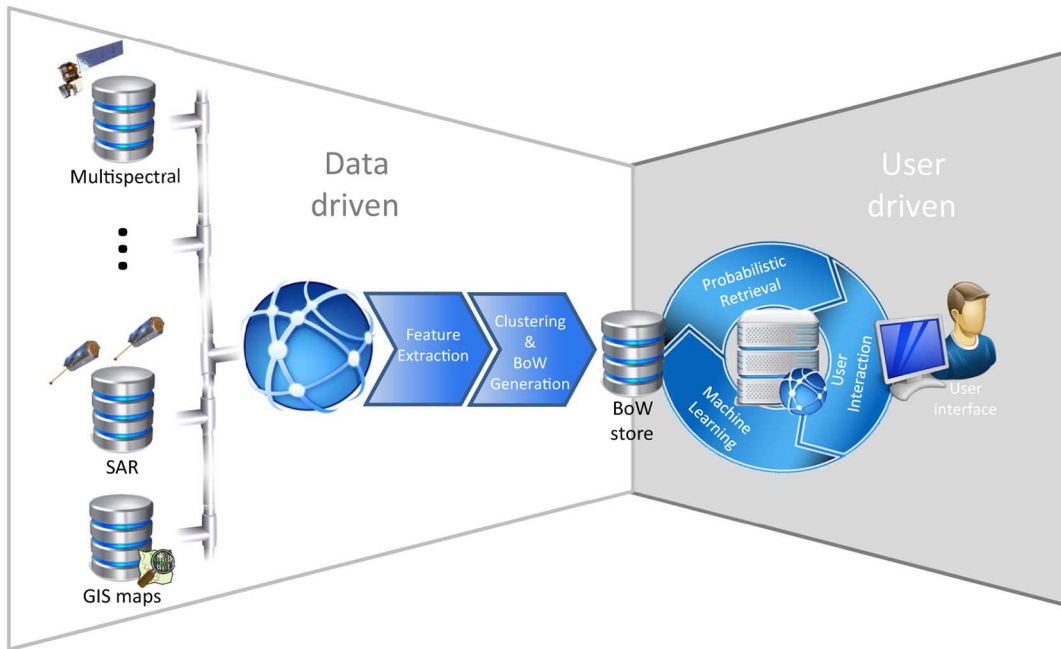


Fig. 2. HDM conceptual design is composed of two main stages: 1) a data-driven initial stage; and 2) the final user-driven stage. The initial modules of the data-driven stage represent the different heterogeneous EO data bases, or GIS repositories, where the data are obtained. The data-driven stage extracts the features, clusters them, and generates the BoW for each image. The user-driven part is composed of the user interaction, the learning, and the probabilistic retrieval processes. The user can introduce positive and negative examples about specific semantics that will allow the system learning. After the learning step, the user-driven cycle will end with a probabilistic retrieval query to obtain the desired images.

using any kind of unsupervised clustering, e.g., k-means. The clustering results, i.e, cluster identifiers, are used for the generation of BoW signatures. At the end of the analysis processes, the word maps, that defines the cluster or word assigned to every pixel in the image, and the calculated BoW signatures $p(\omega_i|D)$ are stored into a database.

In the second stage, the user interaction enters in scene and guides the processes inside an AL loop. This loop is composed of three steps: 1) machine learning; 2) probabilistic retrieval; and 3) user interaction. This stage is real time from the point of view of the user, who expects a relatively fast response of the system to the required actions. Inside the loop, the user interacts via UI introducing positive examples of the label L and negative examples for $\neg L$ allowing the learning of the system. After the interaction step, the user can perform a probabilistic retrieval. If the results are not satisfactory, the user can refine the learning and retry the probabilistic retrieval. On the other hand, if the results are satisfactory, the interaction ends.

Following, a more detailed HDM prototype review is presented, explaining the main functionalities and algorithms used in each module.

IV. FEATURE EXTRACTION

This module extracts at pixel level the image analysis features and map features required by the system to uniquely identify an EO image. For the posterior assumptions of the probabilistic retrieval, the generation of unambiguous stochastic links are required. To achieve this, a full statistical independence of the extracted features θ would be optimal. Thus, a careful selection of features and their coding is necessary to

provide sets of descriptors that are as statistically independent as possible.

In this case, the features used are spectral parameters, such as multispectral features or intensity values; and texture features, as the Weber's law descriptor (WLD) [20]. The system modular design allows to easily add new feature descriptors.

HDM system prototype also relies on the information extracted from existing maps stored in GIS platforms. It is possible to access map information from two different sources. The first one comes from official sources like national governments, government agencies, or supra national organisms. This kind of source is everyday more abundant due to open data and open Government initiatives. The second one is made available by open collaborative projects like OpenStreetMap [21], where the platform users upload geographical information every day and worldwide.

For this prototype implementation, we have used the information available in OpenStreetMap. The layer extraction for this particular initial experiment has been manually done using open source Quantum GIS program [22], but it can be easily automatized in future implementations.

At the end of feature extraction operations, we have available different feature sets.

V. CLUSTERING AND BOW

The statistical independence of the features will help obtaining noncorrelated BoW dictionaries, which in theory, provide more meaningful results when combined. The features obtained from the feature extraction module are used as input for the clustering and BoW module. This module produces two

different outputs for every image and analyzed feature. The first output is the word map that defines the cluster or word assigned to every pixel in the image. The second output is the probability of every word in the image, which is built using the previously calculated word maps. At the end of the data-driven part, the word maps and BoW probabilities are linked together with the analyzed heterogeneous EO images defining a query object in a database.

For the EO images, the clustering of the features relies on the unsupervised K-means clustering algorithm. Clustering processes can take from days to weeks depending on the size of the dataset and the feature length. In the case of GIS maps, the selected layers from OpenStreetMap will represent the cluster identifiers of the map model in the system. In this case, we have chosen seven different classes: 1) water bodies; 2) roads; 3) rail-ways; 4) buildings; 5) urban areas; 6) sport areas; and 7) green areas. The map information is in vector form and, hence, a vector-to-raster conversion is needed. The map rasterization is done using the GDAL library. In order to do the conversion, the raster image must have the same resolution of the EO images with which it will be fused. The resulting raster image is composed of simple integers losing all the semantic meaning of the OpenStreetMap classes. The integer layer is now treated like the output of a clustering process required to generate a word map. Using the word map, the BoW probabilities are calculated. Finally, the word map and the BoW probabilities are stored into the database and linked to the existing query objects created with the EO image analysis.

As stated before, a single feature is in general not enough to generate a meaningful representation of a semantic label. For this reason, even if initially the feature dictionaries are calculated independently, when running the system, a combination of the independent feature dictionaries are used for the learning

$$\omega_i = \omega_{f_1,j} \otimes \omega_{f_2,k} \otimes \dots \otimes \omega_{f_c,z} \quad (1)$$

where ω_i is the combined dictionary, the different f values represent each of the independent feature used in the learning and the subscripts associated to them (j, k, \dots, z) represent the length of that specific dictionary. The length of ω_i will be the product of the independent dictionary lengths.

VI. MACHINE LEARNING

In the following, we present a brief summary of the theoretical concepts of the learning stage based in a naive Bayes classifier [18], [23]. The learning is based on the posterior probabilities of a user-defined semantic label L given an image D expressed as

$$p(L|D) = \sum_i p(L|\omega_i) \cdot p(\omega_i|D). \quad (2)$$

As an alternative, one can apply the Bayesian theorem

$$p(L|D) = p(L) \cdot \sum_i \frac{p(\omega_i|L) \cdot p(\omega_i|D)}{p(\omega_i)} \quad (3)$$

where $p(L)$ is the prior probability of the semantic label L , $p(\omega_i|D)$ are the probabilities of the words in a given image,

$p(\omega_i|L)$ denotes the probabilistic links of the words with a label, which can be expressed as the probability of the words updated with the examples defined by the user. Finally, $p(\omega_i)$ is the prior of the words ω_i given by

$$p(\omega_i) = \sum_L p(\omega_i|L) \cdot p(L) \quad (4)$$

where the labels are restricted to L and $\neg L$ and the probabilistic link $p(\omega_i|L)$.

At this point, the computation of $p(\omega_i|L)$ and $p(\omega_i|\neg L)$ should be considered. Once these terms are known, $p(L|D)$ can be calculated using (3) and (4). In order to calculate these probability links, we will make use of the user inputs by means of training samples. The training samples can be positive, represented by the presence of the label L , or negative referring to the absence of L and defined as $\neg L$. For sake of simplicity, the next equations present the calculation of $p(\omega_i|L)$, but they are applied to $p(\omega_i|\neg L)$ in the same way. Therefore, we define T as a set of user-provided positive training data in the form of $T = \{N_1, \dots, N_r\}$ with N_i as the number of occurrences of ω_i and r as an index of the existing words. T presents a multinomial distribution that can be parametrized via $\phi = \{\phi_1, \dots, \phi_r\}$. As introduced in [6], and widely described in, [24] and [25], we can express our desired probability as

$$p(\omega_i|L) = E[\phi_i] = \int \phi_i p(\phi|T) d\phi_i \quad (5)$$

where $p(\phi|T)$ is modeled as a Dirichlet distribution

$$p(\phi|T) = Dir(\phi|\alpha) \quad (6)$$

and α is a hyperparameter vector which represents the user interaction (i.e., introduction of training examples by the user). It has the same dimension as the used BoW dictionary, and is initialized to one, $\alpha = \{1, 1, 1, \dots, 1\}$. One of the Dirichlet model characteristic is the property to perform the learning incrementally. Moreover, a Dirichlet distribution is the conjugate prior of a multinomial distribution in Bayesian statistics. Thus, new user interactions can update the posterior probability by means of

$$p(\phi|T) \sim Dir(\alpha_1^k + N_1^{k+1}, \dots, \alpha_r^k + N_r^{k+1}) = Dir(\alpha^{k+1}) \quad (7)$$

where k refers to the user interaction, N_i^{k+1} are the new training examples, and the updated hyperparameters α_i^{k+1} are defined by the following expression:

$$\alpha_i^{k+1} = \alpha_i^k + N_i^{k+1}. \quad (8)$$

Once the hyperparameters and the update procedures are defined, we can rewrite (5) as

$$p(\omega_i|L) = \frac{\alpha_i^{k+1}}{\sum \alpha_i^{k+1}}. \quad (9)$$

Since $p(\omega_i)$, expressed in (4), is the sum of L and $\neg L$, using the negative samples introduced by the user, we define another hyperparameter vector set for the required $p(\omega_i|\neg L)$ calculation. With these hyperparameter sets, we are finally able to calculate the posterior probabilities $p(L|D)$, defined in (3), and proceed to the probabilistic search.

A. Assumption of Feature Probability Independence

The original KIM implementation assumes the full statistical independence of the features and the resulting clusters. Thus, the calculation of the stochastic link can be performed by a simple multiplication of probabilities

$$p(\omega_i|L) = p(\omega_{f_1,j}|L) \cdot p(\omega_{f_2,k}|L) \cdot \dots \cdot p(\omega_{f_c,z}|L) \quad (10)$$

where c is the number of feature models used in the learning, f identifies the feature model, and the associated subscript represents its length. The same statistical independence assumption is made for the joint probability of the words in a given image $p(\omega_i|D)$.

This approach was already computationally fast, with the imposed restriction of the use of only two different feature models in the learning stage: spectral and texture features were used to ensure the required statistical independence. The restriction was imposed due to the computational complexity from the calculations of $p(\omega_i|L)$ and $p(\omega_i|D)$ which can be represented as multidimensional matrices, where each element refers to the probability of occurrence of a certain word combination. Therefore, every additional feature model increases the dimensionality of these matrices by one. As a consequence, the number of operations are multiplied by the number of words of each feature model. Defining n as the number of operations required for the independent calculation of the posterior probabilities for each feature model, and c being the identifier of the model, we obtain the computational complexity as

$$O = n_1 \cdot n_2 \cdot \dots \cdot n_c. \quad (11)$$

The different sizes of the dictionaries in ω_i are relatively small in comparison with the total amount of operations needed for the calculation of the posterior probabilities. Therefore, the different numbers of operations n_c can be equalized to $n_c = n$. This results in a final polynomial complexity of the algorithm, increasing with c

$$O(n^c). \quad (12)$$

B. Assumption of Posterior Probability Independence

The complexity of the KIM algorithm should be reduced since its algorithm, based on the statistical independence of the features (10), is not fast enough in a high resolution EO Big Data scenario. Our proposed approach extends the statistical independence assumption from the features, which has been proved valid in [26] and [27], to the posterior probabilities. The proposed approach is derived from the belief that the statistical independence can be inherited if the extracted features from the original data are independent. In this case, the proposed statistical independence assumption is defined as

$$p(L|D) = p(L|D)_1 \cdot p(L|D)_2 \cdot p(L|D)_3 \cdot \dots \cdot p(L|D)_c \quad (13)$$

where $p(L|D)$ is the product of the individual posterior probabilities of each feature model, and c is the total number of feature models used.

As stated in (3), the posterior probability calculation requires the knowledge of $p(\omega_i|L)$ and $p(\omega_i|D)$. Assuming the posterior probability independence, we will treat each feature model dictionary independently. Furthermore, we avoid the calculations of the joint probabilities and the iterations over multidimensional representations of $p(\omega_i|L)$ and $p(\omega_i|D)$. Thus, it is possible to greatly reduce the number of operations required for the calculation of $p(L|D)$. Moreover, the computational complexity of the new algorithm is simplified. With our new statistical independence assumption the complexity can be determined as the addition of the different feature model complexities

$$O = n_1 + n_2 + \dots + n_c. \quad (14)$$

Simplifying the different complexities to n , as in the previous case, and assuming c is not meaningful when compared to n , the complexity changes from polynomial to linear as follows:

$$O(n \cdot c) = O(n). \quad (15)$$

The complexity reduction due to the new statistical independence assumption results in a huge acceleration of the required computational effort. This acceleration can be used for the inclusion of new features models in the learning stage. Since more feature models mean an extension of the possible combinations of words ω , and in consequence extended discrimination capabilities, this will be useful in more complex user semantics definition processes.

VII. PROBABILISTIC RETRIEVAL

For the last decades, the machine learning community has used multiple feature distances for the classification and retrieval of different multimedia assets [28]. In [29], once the information from images is captured in a feature set, two different ways to endow images with meaning are presented. The first compares the feature set with the elements in a training set, leading to conditional probabilities that sketch an interpretation of the image, but does not determine it completely. This approach is described in Section VII-A. The second approach relies exclusively on the feature set to generate visual signatures and compute the similarities. Examples of this approach can be found in [3]. In Section VII-B, we propose a modification of the classical approach, consisting in the calculation of the similarity between the elements contributing to the posterior probability of the query image and the image signatures obtained with the BoW.

A. Retrieval Based on the Posterior Probability Value

This retrieval method is the one originally implemented in KIM. The method proposed a probabilistic retrieval that relies on the $p(L|D)$ value of the images in the database. Thus, the $p(L|D)$ of every image in the database is calculated and ranked by its value. The images with a higher probability of containing the user requested semantic label L appear in the initial positions of the ranking.

B. Retrieval Based on Similarity Metrics

A classical retrieval by similarity relies on the image signatures computed exclusively from features. In our system, the visual signatures are represented by the BoW probabilities $p(\omega_i|D)$. Thus, a classical similarity retrieval would include the similarity calculation between the stored $p(\omega_i|D)$.

Our contribution to the probabilistic retrieval modifies the classical approach calculating the similarity distance among the $p(\omega_i|D)$, BoW signature of each element in the database, and the elements used for the calculation of $p(L|D)$ according to (2). By doing so, we introduce the user-specific semantics into the similarity computation.

The use of similarities or distances for the retrieval allows us to introduce a new parameter in the retrieval process, namely the distance metrics. We have implemented a set of different metrics to calculate distances, d

1) *Euclidian:*

$$d_E = \sqrt{\sum_i ((p(L|\omega_i) \cdot p(\omega_i|D)) - p(\omega_i|D))^2}. \quad (16)$$

2) *Kullback–Leibler:*

$$d_{KL} = \sum_i p(L|\omega_i) \cdot p(\omega_i|D) \cdot \ln(p(L|\omega_i)). \quad (17)$$

3) *Kullback–Leibler Symmetric Variant:*

$$d_{KLS} = \sum_i p(L|\omega_i) \cdot p(\omega_i|D) \cdot \ln(p(L|\omega_i)) + \sum_i p(\omega_i|D) \cdot \ln\left(\frac{1}{p(L|\omega_i)}\right). \quad (18)$$

4) *Jensen–Shannon Divergence:*

$$d_{JSD} = \frac{1}{2} \cdot \sum_i p(L|\omega_i) \cdot p(\omega_i|D) \cdot \ln\left(\frac{p(L|\omega_i) \cdot p(\omega_i|D)}{M}\right) + \frac{1}{2} \cdot \sum_i p(\omega_i|D) \cdot \ln\left(\frac{p(\omega_i|D)}{M}\right) \quad (19)$$

where $M = 1/2 \cdot (p(L|\omega_i) \cdot p(\omega_i|D) + p(\omega_i|D))$.

5) *Manhattan:*

$$d_M = \sum_i |(p(L|\omega_i) \cdot p(\omega_i|D)) - p(\omega_i|D)|. \quad (20)$$

6) *Chebychev:*

$$d_{Ch} = \max_i \{|(p(L|\omega_i) \cdot p(\omega_i|D)) - p(\omega_i|D)|\}. \quad (21)$$

The availability of different metrics is another resource that the user can exploit in order to improve the image retrieval. As we will present in Section XI, the use of a specific distance metric can be useful for certain user concepts.

VIII. UI

The UI, shown in Fig. 3, is presented as a QbE interface where the user can load an image from the repository. The user can select the example image directly navigating the repository or simply by selecting one of the 20 images that randomly are shown in the right part of the UI. These random images can be refreshed at any time just pressing the “Random” button. The first main canvas is used to represent the query example. Depending on the dataset and the features selected for the learning, it is possible that the query element contains more than one analyzed EO image, see Section IV. In these cases, the query canvas representation can be switched between these source images. The second main canvas, located in the center of the UI, represents a posterior probability map (PPM) defined as the posterior probability ratio of each pixel in the image

$$PPM = \frac{p(L|d_n)}{p(L|d_n) + p(-L|d_n)} \quad (22)$$

where d_n are the individual pixels in D , $p(L|d_n)$ is the posterior probability of the label L given a pixel d_n and $p(-L|d_n)$ the posterior probability of $-L$ in the pixel.

The PPM is updated with every user input providing an useful interactive tool to check the validity of the learning process. Moreover, the UI also implements different drop lists where the user can select the following:

- 1) *System algorithm:* The user can select between KIM or HDM algorithm.
- 2) *Retrieval method:* It is possible to select the probabilistic retrieval method based on posterior probability value or the one based on similarity metrics.
- 3) *Feature models:* The user can choose different combinations of feature models which will be used in the learning-retrieval processes.
- 4) *Similarity metric:* If the probabilistic retrieval based on similarity metrics is selected, a list for selecting the desired metric is enabled. The available metrics are listed in Section VII-B.
- 5) *Image source:* Certain feature model combinations have different EO image source. For these cases, the user can switch at any moment the image source shown in the main canvas. This option can be helpful during the learning process in order to improve the quality of the input introduced by the user.

After every query, the retrieved images are ranked under the main canvas. Those ranked images can be clicked so that they can be loaded as example image for continuation of the training. Finally, there is the option to save, load, and update the user-defined labels L and the associated training hyperparameters.

IX. USER INTERACTION

In a common search procedure, the PPM starts all in grey, representing the unknown state of every pixel in the image due to the lack of positive or negative examples. With every provided example N_i^{k+1} , the hyperparameters α_i^{k+1} and the stochastic link $p(\omega_i|L)$ are updated. As a consequence, the posterior probabilities $p(L|d_i)$ are also updated and with them the

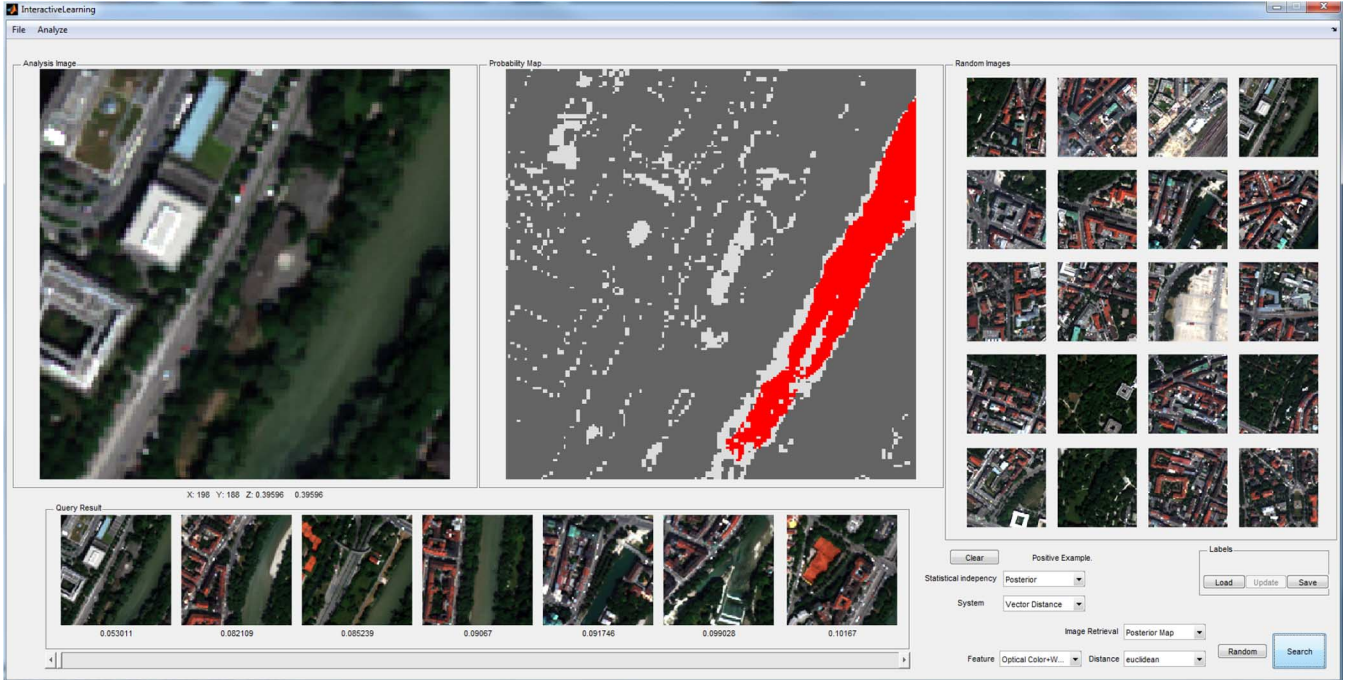


Fig. 3. System UI. The first main graphical canvas represents the query image example, the second one the posterior probability, $p(L|d_n)$, values of each pixel. On the right, a group of random images from the database are shown. At the bottom under the main canvases the query results are shown. On the right, under the random images, the parameter selection drop lists, label load/save, and search buttons are shown.

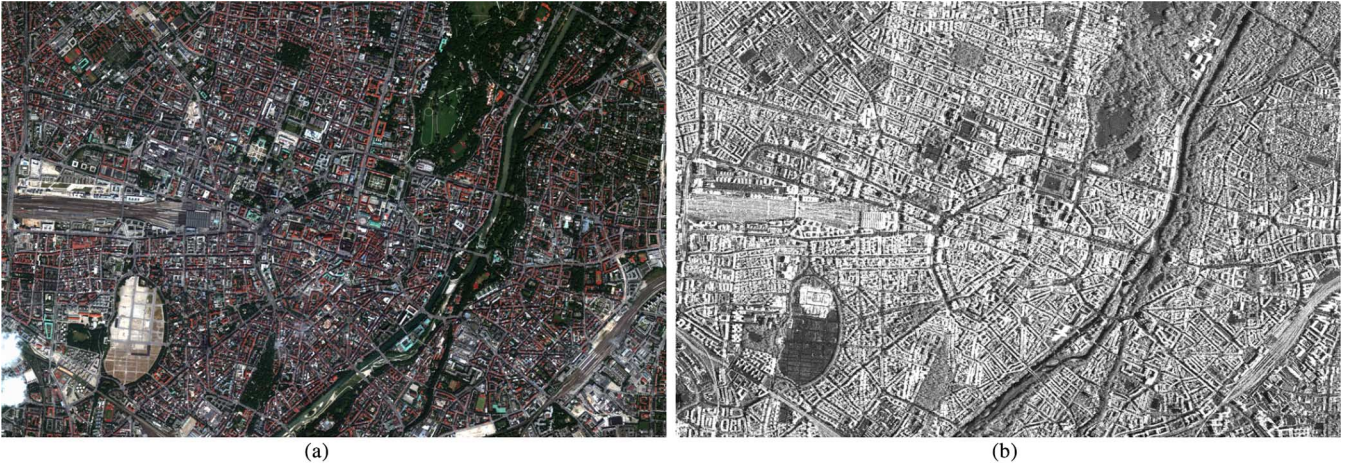


Fig. 4. Multispectral-SAR case scenario of Munich, Germany. (a) WorldView-2. (b) TerraSAR-X.

PPM, defined in (22), that is shown to the user. Black pixels will represent the low probability and white ones the high probability. Once the probability of a pixel is over 0.9, the pixel is highlighted in red.

At any moment, the user can perform a query to check the retrieved results. By checking the results, it is possible to refine the learning process introducing, for instance, negative examples over an image retrieved in the first positions, but which does not contain the label L . As a direct response of this negative example, the image will be penalized in the next search appearing in a lower ranking position.


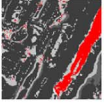





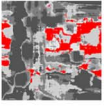




During the learning retrieval process, the user can decide to try one of the different approaches implemented for probabilistic retrieval. Using the probabilistic retrieval method based on

TABLE I
SYSTEM QUERY RUN-TIME FOR DIFFERENT STATISTICAL ASSUMPTIONS,
QUERY RANKING TYPES, AND FEATURE MODEL NUMBERS

	Search Proc.	1 Model (s)	2 Model (s)	4 Model (s)
Feature independence (KIM)	Posterior probability	0.62	6.39	2354
	Sim. metric JSD	0.045	0.15	43.5
Posterior independence (HDM)	Posterior probability	0.155	0.196	0.31
	Sim. metric JSD	0.042	0.126	34.61

The first row is used as a threshold and corresponds to an emulation of the original KIM implementation. The difference among the four models using the new statistical assumption is four orders of magnitude.

TABLE II
SYSTEM QUERY RESULTS

Exp.	Label	Query example	Alg.	Retrieval method	Posterior map	Retrieved patches	Error position	Error
A.1	River		HDM	Posterior probability			1	
			HDM	Sim. metric KL			11	
A.2	Railway		KIM	Posterior probability			10	
			HDM	Posterior probability			13	

First experiment shows a better performance of the similarity metric retrieval method in initial learning stages. The second experiment shows how the simplification of the calculation processes obtained with the HDM algorithm does not affect the quality of the retrieved results. Moreover, for some cases the query results are even better.

similarity metrics, it is also possible to try one of the several implemented distance metrics described in Section VII. Once the user agrees with the semantic label definition, it can be stored in the database to be reused in further analysis.

X. APPLICATION SCENARIOS

In order to test the applicability of the system and the implemented new algorithms, we present three different scenarios. The analysis of these scenarios is only possible due to the speed-up achieved with the new learning algorithm, which allows the introduction of a higher number of feature models. In the first scenario, called multispectral-SAR fusion, the goal is to demonstrate the system speed-up and performance in comparison with the original KIM implementation for urban assessment. The second scenario, multitemporal-SAR fusion, uses the system with image time series for change detection applications. In the final scenario, multispectral-SAR-map fusion, we combine the classical image analysis feature models with a model extracted from map information stored in a GIS server.

A. Case: Multispectral-SAR Fusion

For validating the system, we have chosen Munich city multispectral images from WorldView-2 and SAR images from TerraSAR-X both with 1.25-m pixel spacing, covering an area of 24 km², as shown in Fig. 4. The size of the total scene is 4890 × 3202 pixels cut into tiles of 200 × 200 pixels, with a total number of 500 tiles. The clustering and BoW generation of this dataset resulted in 256 words for the intensity feature and 19 words for the WLD texture feature in the multispectral image, along with 124 words for the intensity and 8 words for the texture features of the SAR image. Summarizing, each

database object is composed of one multispectral patch, one SAR patch, four word maps, and the associated probabilities.

The evaluation of the system and algorithm performance is done by measuring the time required for the completion of a query and the quality of the query results. We compare the execution time of the original KIM algorithm with the modified HDM algorithm presented in Section VI-B.

The first experiment of this scenario calculates the query processing speed of the system for KIM and HDM algorithms using the two available retrieving methods presented in Section VII. The feature model combination used are: multispectral intensity for the unic model case, multispectral intensity and texture when two models are used; and multispectral and SAR intensity and texture in the four model tests. Table I summarizes the obtained results. In the case of the KIM algorithm, the improvement of speed using similarity metric retrievals amounts to one or two orders of magnitude compared to the posterior probability approach. When comparing KIM and HDM algorithm performances, HDM turns out to be faster. The HDM similarity metric retrieval method performs faster than KIM but in the same order of magnitude. In contrast, using the posterior probability retrieval method, the HDM performs four orders of magnitude faster than the original KIM implementation for the four feature model case.

In a second experiment, we define a fixed learning process based on positive and negative examples over the same pixels in the same images. The first stage of this experiment, A.1 in Table II, shows the initial results and the first error (i.e., the first misclassification result) of the similarity metric-based retrieval versus the posterior probability ranking methods when looking for the user query *river*. The four-feature models are used in the learning stage. After a few user interactions, the query output shows an improvement of the results using the similarity metric retrieval method. For this case, the patches shown

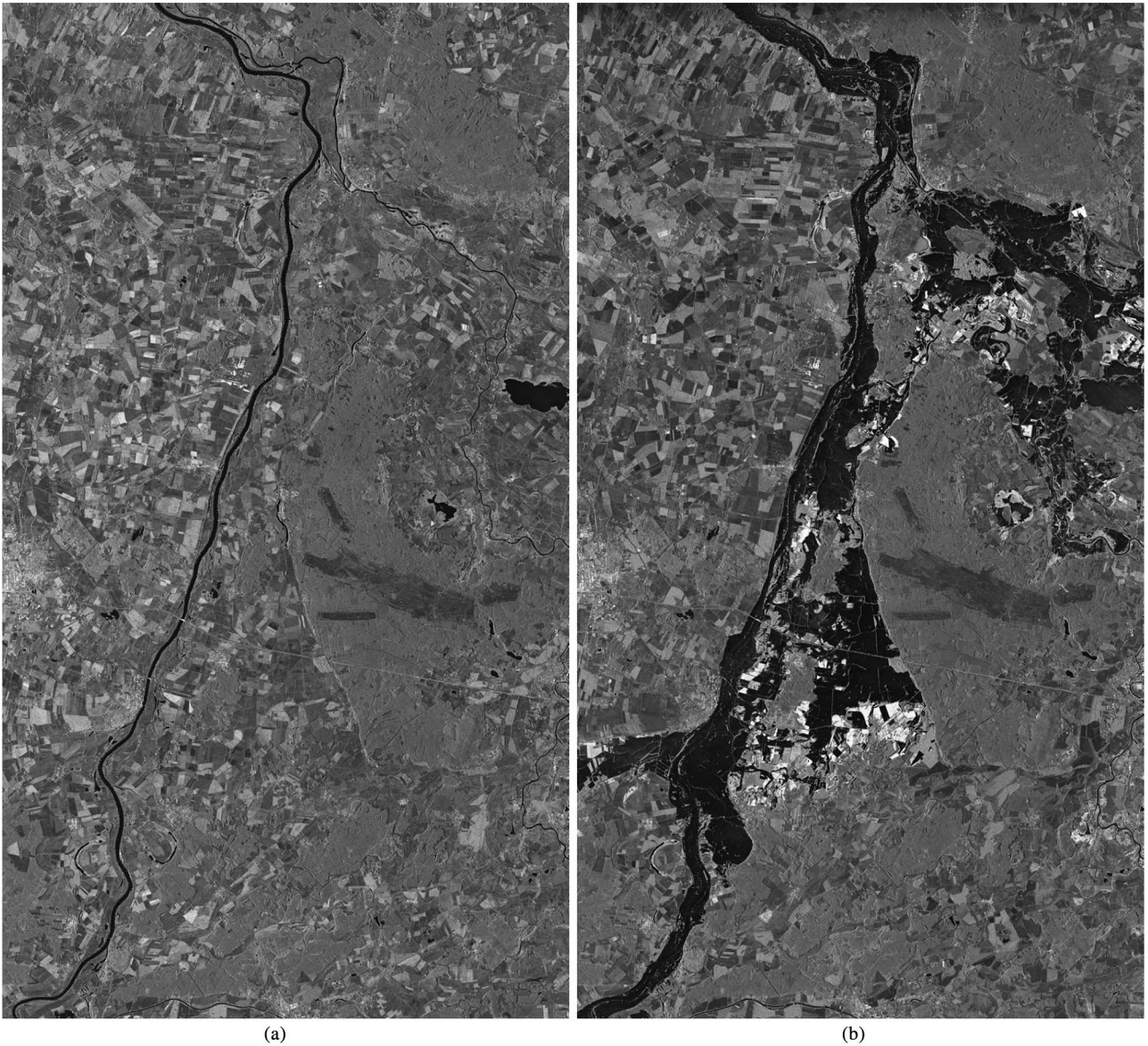


Fig. 5. TerraSAR-X images of Elbe river years before and during the 2013 flooding. The flooded areas can be seen in black. (a) June 26, 2008. (b) June 15, 2013.

in the initial positions match the query image; meanwhile, the posterior probability retrieval method is still insufficiently trained to provide correct results.

In the second part of this experiment, we evaluate the final query response using the feature statistical assumption implemented in KIM and the new HDM algorithm. We show the first ranked images and the first misclassified element, with its position in the rank. In the results, shown in A.2 of the Table II, the user searches for *railways*. We can observe how the HDM algorithm provides also good results. Moreover, in some cases, like in the one shown in the experiment results, the first error appears even later, at position 13th.

B. Case: Multitemporal-SAR Fusion

For the second application scenario, a sequence of two TerraSAR-X images has been chosen in order to show the system capabilities to detect changes in an image time series. In

this case, the query example involves at the same time feature models generated from both SAR images. The images cover a part of the Elbe river course in Germany, see Fig. 5. This region suffered a severe flood during 2013. The images correspond to an initial acquisition on June 26, 2008, and a second one during the flooding on June 15, 2013. Both images have the same characteristics, they correspond to a stripmap level 1B product with horizontal polarization and 5.88-m azimuth and ground range resolutions. The size of the scene is $11\,453 \times 20\,528$ pixels with a pixel spacing of 2.75 m and a covered area of 1778 km^2 . Each image has been cut in small tiles of 200×200 pixels, resulting in a total of 5814 patches. Like in the previous case, the system manages pixel-based feature clustering results. The four features used are the intensities, from which we obtain 56 words, and WLDs, 31 words, from both SAR images. For this dataset, each database object is composed of two SAR image patches, four word maps, and the associated probabilities. In this case, we assume the statistical independence based on the

TABLE III
MULTITEMPORAL-SAR FUSION CASE SCENARIO

Exp.	Label	Query example	Year	Retrieval method	Posterior map	Retrieved patches				Error position	Error
B.1	Flood		2008	Posterior probability						1300	
			2013								
B.2	No flood		2008	Posterior probability						1771	
			2013								
B.3	Crop change		2008	Posterior probability						1200	
			2013								

B.1 experiment shows the case where the user searches for flooded regions. The experiment *B.2* shows the case in which not flooded images are requested. In the last experiment of this case scenario, *B.3* images with crop change during the time series are requested.

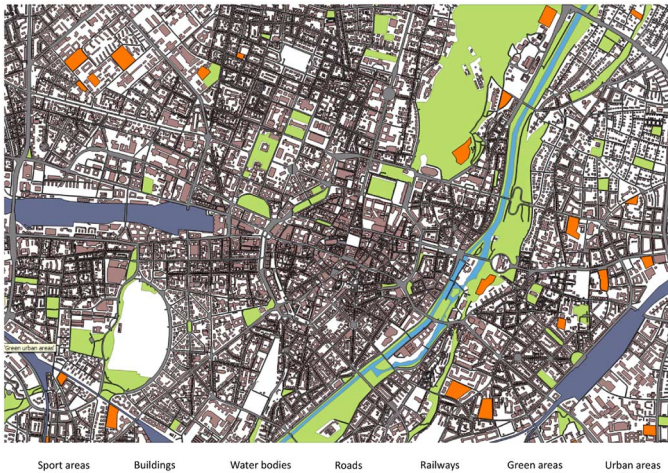


Fig. 6. Raster map with several classes extracted from OpenStreetMap. Munich city, Germany.

large time interval between acquisitions, 5 years. To verify the statistical independence, we computed a similarity map based on the normalized compression distance (NCD) [30], obtaining high dissimilarity values. This experimental approach was employed since the analytical verification of statistical independence is a highly complex problem [31]. In the next three experiments, we retrieve all the patches in the database. The ranking is done by using the posterior probability value, first to get the patches with a higher probability of containing the semantic concept defined by the user.

The first experiment, *B.1* in Table III, aims the search of flooded areas in the scene, where a flooded area is represented as an object in the database containing areas with no water in 2008 and which are covered by water in 2013. After giving just one positive example, the first error (i.e., the one corresponding to an image with no flooding) appears at position 132nd. Continuing with the training, misclassifications start to be common only after position 1300th.

The second experiment of the multitemporal scenario, *B.2* aims to retrieve nonflooded patches. This includes images with no change at all (e.g., permanent course of the river) and images with changes not related to the flooding (e.g., crop changes). The first patch with severe flood appears at position 338th just with an unique positive example. Following the tuning of the learning process with more positive and negative examples, the error proliferation starts at 1771st position and they start to be regularly ranked after position 4600th.

In the third experiment *B.3*, the user retrieves images with agricultural fields that have changes in the the crop. The initial query, with only one positive example, provides correct results until position 33rd. Continuing the learning, the first misclassification goes backward to position 364th and their appearances become more regular after position 1200th.

C. Case: Multispectral-SAR-Map Fusion

In this scenario, the system is tested with the same dataset used in Section X-A, i.e., optical bands of a multispectral image

TABLE IV
MULTISPECTRAL-MAP FUSION CASE SCENARIO

Exp.	Label	Query example	GIS	Retrieval method	Posterior map	Retrieved patches				Error position	Error
C.1.1	Small path		No	Posterior probability						1	
			Yes							1	
C.1.2	Small path		No	Posterior probability						5	
			Yes							17	
C.2.1	Dry river border		No	Posterior probability						1	
			Yes							6	
C.2.2	Dry river border		No	Posterior probability						1	
			Yes							22	
C.2.3	Dry river border		No	Similarity metric KL						9	
			Yes							22	

In the first experiment *C.1*, the user searches for *smallpaths* surrounded by vegetation. *C.1.1* shows the results after the first user example. *C.1.2* shows the results after two positive and one negative example. The last experiment, *C.2*, aims to search for the user-defined concept *dryriverborder*. In *C.2.1* and *C.2.2* the system is unable to learn the user concept if the map model is not used. For the case with map layer the retrieved images contains the defined label. *C.2.3* shows that when using matrix distance search, even without the map model, the concept learning is possible and for the case with map model the learning process tends to be faster.

(WorldView-2) and a SAR (TerraSAR-X) image (see Fig. 4). Additionally, as GIS map feature model, we generate a raster image using information from the OpenStreetMap collaborative project (see Fig. 6). From this map feature model, we obtain seven different words corresponding to the extracted classes in Section IV. Thus, the database objects are composed of one multispectral patch, one SAR patch, four word maps with the associated probabilities from the EO images, and a word map with the associated probabilities from the GIS map.

The first experiment intends to demonstrate the acceleration of the learning process due to the insertion of a generic map feature model. The features used in this experiment are intensity

and WLD features from both images and the GIS map. The retrieval method is the one based on the total sum of the posterior probabilities. In this experiment, the user looks for tiles with *small paths* in parks or gardens. The first stage, *C.1.1* in Table IV, shows the speed-up of the learning process since the first positive example. The first row presents unsatisfactory results obtained without the inclusion of the map feature model. In contrast, including the map, the first ranked images contain the user-defined concept. Here, due to the optical intensity feature model, white pixels are highly pushed up in the ranking. The use of the posterior probability-based retrieval method, brings the tiles containing a bigger amount of white pixels

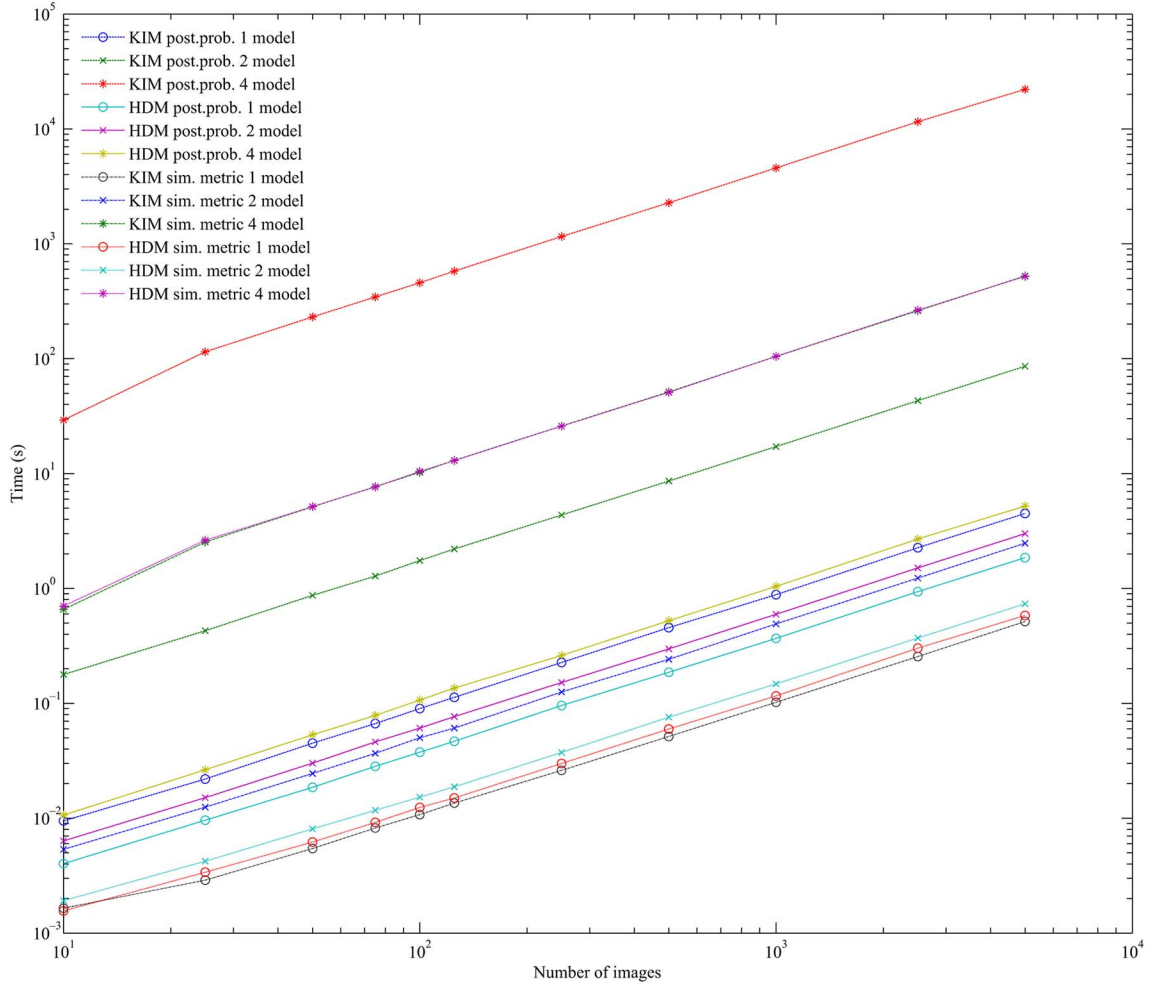


Fig. 7. Query computation time curves. It is clearly visible the difference in orders of magnitude between retrieval time using KIM and HDM algorithms. The complexity of the KIM algorithm limits in practice the number of feature models in the learning stage. The used of HDM allows us to convert the learning process into a linear complexity problem. This enables a real-time response to every user interaction and for the database querying process. The similarity metric retrieval method is the most efficient one up to the combination of two models. After that, the posterior probability retrieval method performs better.

(e.g., tiles containing clouds or places under construction) to the top of the ranking. Nevertheless, for the query including a GIS feature model, the inclusion of extra information related to roads, adds an extra discrimination capability. This enables, from the very beginning, the retrieval of positive matches in the top of the rank.

C.1.2, in Table IV, shows the first top four positions of the rank after adding one more positive and one more negative examples. The initial results are good in both cases. However, in the case without a map feature model, the first error appears at the 5th position. Meanwhile, in the case with a GIS feature model, it appears at position 17th.

The second experiment shows the system performance for the discovery of more complex classes like *dry river borders* with and without the addition of the feature model from OpenStreetMap. The query method in the two first stages *C.2.1* and *C.2.2* is based on the posterior probability. The third stage *C.2.3* uses the similarity metric retrieval in order to show the importance of the chosen retrieval method after the same learning process.

As in the previous experiments, *C.2.1* shows the response of the system after one initial positive example. Only the

experiment with a GIS map feature model shows initial good results. Continuing the AL, *C.2.2*, the case without the map feature model is unable to learn the concept, providing only a correct tile in position 16th of the rank. However, for the query including the map feature model, the learning improves. After three positive and two negative additional examples, the first wrong result moves from the initial 6th position, in the *C.2.1*, to the 22nd.

In the last stage of this experiment, *C.2.3* in Table IV, we compare the retrieved results following the same learning process of *C.2.2*, but using now the similarity metric retrieval method. In contrast to *C.2.2*, even not using a GIS feature model, it is possible now for the system to learn the user concept and obtain positive results. The first error appears in this case at the 9th position. In the second query, a map feature model is used, obtaining results with the same quality with a smaller number of interactions.

XI. PERFORMANCE EVALUATION AND DISCUSSION

More general system results can be extracted by manually annotating the previously introduced datasets. The annotation

TABLE V
SYSTEM QUERY RESULTS

User concept	Feature models	Images retrieved	Metric	Precision (%)	Recall (%)	Accuracy (%)	F1 measure (%)
River	Multispectral intensity & WLD, SAR intensity, & WLD	25	Euclidian	44	45	92	45
			Kullback–Leibler	72	75	96	73
			Kullback–Leibler S.	76	79	97	78
			Jensen–Shannon	72	75	96	78
			Manhattan	76	79	97	78
			Chebychev	80	83	97	81
Turquoise roof	Multispectral intensity, WLD	100	Euclidian	45	40	66	42
			Kullback–Leibler	84	74	88	79
			Kullback–Leibler S.	53	47	70	50
			Jensen–Shannon	80	70	85	74
			Manhattan	50	44	69	47
			Chebychev	41	36	64	38

First test defines the user concept *river* using a four feature models in the learning process ranking 25 tiles and using all the implemented metrics. The results show the Chebychev metric outperforms the others in overall. The precision 80%, and recall 83%, are at least a 4% better than the rest of the metrics. The second experiment presents the first one hundred query results for a *turquoise roof* user concept using in the learning stage the two feature models from the multispectral image, intensity and WLD texture. Kullback–Leibler and Jensen–Shannon divergence outperform greatly the rest of the metrics. In this case, Kullback–Leibler outperforms the rest of the metrics with an 84% of precision, 74% of recall.

process involved the generation of a record with the presence of different semantic concepts over the whole datasets. Due to the highly time-consuming nature of the annotation task, it was possible to calculate quantitative statistical parameters only for the concepts presented in this section.

Extending the first experiment of the multispectral-SAR case scenario, we can see in Fig. 7 the system response with different configurations retrieving several dataset sizes. It is possible to detect that for an unique model and for the fusion of two models, the retrieval based on similarity metrics performs faster than the posterior probability retrieval. The reasons are two: first, we avoid the calculation of the posterior probability on the entire database and compute it just on the query image; and second, the dictionary sizes for the distance calculations are short enough to ensure a fast computation over all the elements of the database. However, this tendency changes when the number of feature models to fuse is more than four. In this case, the number of elements in the dictionary increases to a point where the cost of calculating the posterior probabilities in comparison of the computation cost to calculate the distances is trivial. This is exactly the reason why the computation times for four feature models using similarity metric retrieval are alike for both KIM and HDM.

Summarizing, we can say that the distance metric retrieval method is the most efficient one up to the combination of two models. If more feature models are used, the posterior probability value-based retrieval performs faster. This can be explained, first, due to the simplification in the posterior probability calculation over the whole database. Second, because the subsequent ranking of scalar values is less expensive than the one based on similarity metrics.

To validate the introduced probabilistic retrieval based on similarity metrics, different queries were performed using the distance metric as a parameter instead of using a fixed unique metric. Table V shows a summary with the best query results

for two different semantic labels with different combination of feature models. Specifically Table V provides precision, recall, accuracy, and F_1 measures, i.e, the equally weighted harmonic mean of precision and recall. A detailed explanation of the measurements can be found in [32].

In the first test, we define the user concept *river* using the four-feature model employed for the experiment in Section X-A in the learning process. The first ranked 25 tiles are retrieved using all the implemented metrics. The results show the Chebychev metric outperforming the rest of the metrics. The precision 80%, and recall 83% are at least a 4% better than the rest of the metrics. The obtained accuracy is 97% with a value of F1 measure of 81%, 3% better than the second best metric. The second experiment presents the query results for a *turquoise roof* user concept using in the learning stage the two feature models from the multispectral image, intensity and WLD texture. The number of images retrieved using the similarity metric method is 100. It is remarkable how Kullback–Leibler and Jensen–Shannon divergence outperform greatly the rest of the metrics. In this case, Kullback–Leibler provides a 84% of precision, 74% of recall.

Finally, we have extended the multitemporal-SAR case scenario to present more exhaustive quantitative measures of the retrieved results. In Fig. 8 we provide precision, recall, accuracy, and F_1 measure for different user-defined concepts. The graphics show how the different measures vary depending on the number of retrieved tiles from the database. In almost all the cases, the precision of the system for retrieved tiles from 1 to 1000 remains over the 90%. The recall increment varies depending on the amount of images in the database annotated with the concept. The total amount of tiles is 5814, from which 1409 are annotated with the *flooded* label, 4405 tiles with the *nonflooded* label, 1930 tiles annotated as *nonchanged*, and 2474 tiles as *crop change*.

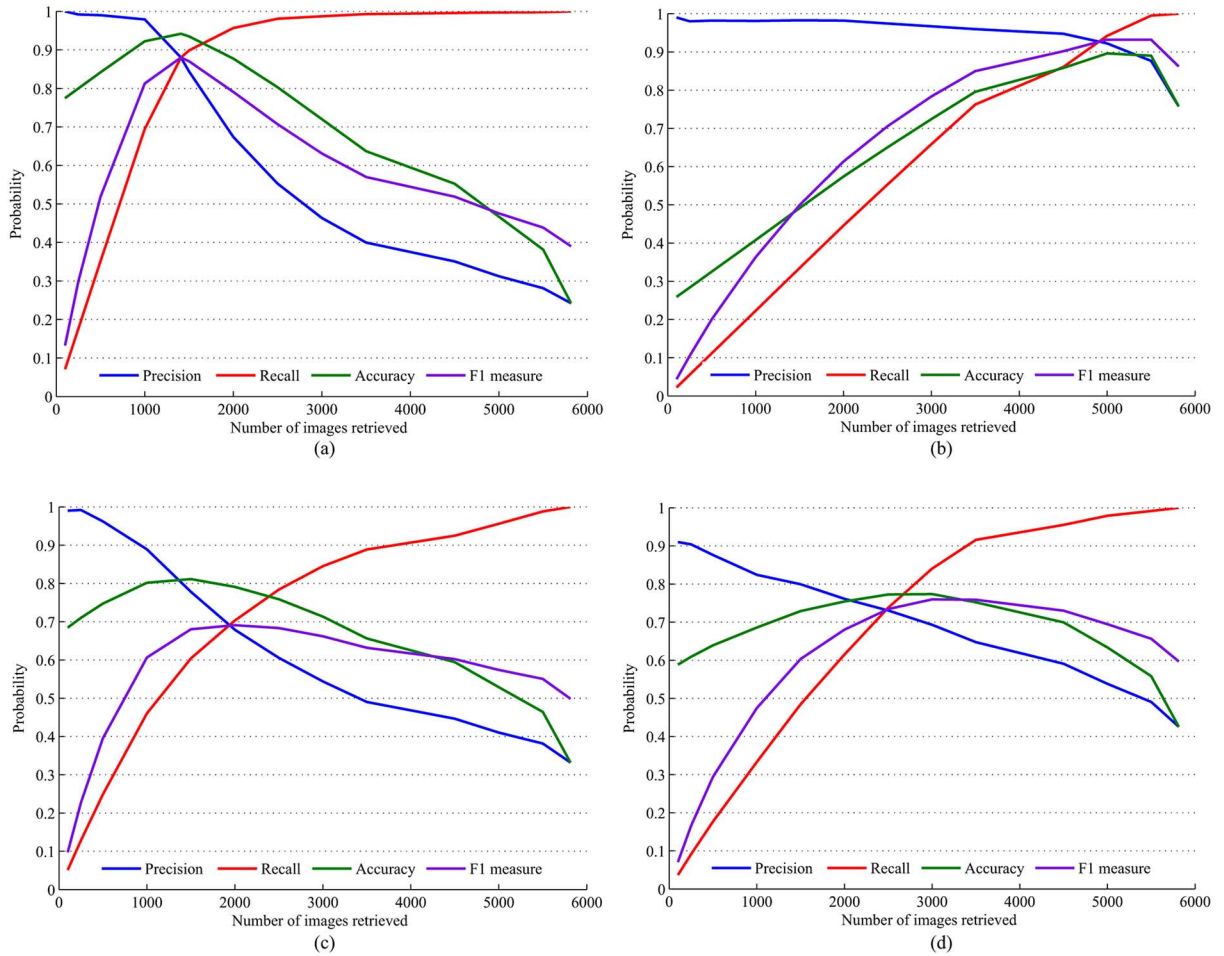


Fig. 8. System quantitative results for the retrieval of different user-specific semantics. (a) Retrieval of flooded images. (b) Retrieval of nonflooded images. (c) Retrieval of images with no changes. (d) Retrieval of images with crop changes.

XII. CONCLUSION

We have presented an HDM prototype inspired by and following the main concept summarized in Section II and previously implemented in the KIM system in [6]. The HDM enhances the original KIM system overcoming the two-model limitation. HDM introduces a faster AL algorithm modifying the required statistical independence from the features to the posterior probabilities. The obtained speed-up allows the introduction of new feature models in the learning stage and the definition of more complex user semantics. The acceleration can also open new ways for KIM systems to Big Data scenarios.

For comparison purposes, we reimplemented the original KIM method, and based on it, we introduced new search methods and theoretical probabilistic assumptions which may outperform in speed the previous one by various orders of magnitude. The proposed probabilistic search method based on the distances between the elements used for the calculation of the posterior probabilities and image BoW in the database performs better for weakly defined labels. However, for two different reasons, this search approach cannot replace completely the approach based on total posterior probabilities. First, the posterior probability-based retrieval yields the tiles with a higher

probability of containing the required user semantics. Second, due to the simplicity of the scalar ranking of posterior probabilities, this approach performs faster for scenarios with a feature model number greater than two. The latter fact is due to the increase of the computation cost of the similarity distances with each extra feature model.

We experience a considerable speed-up of the learning algorithm by introducing the assumption of posterior probability statistical independence, which does not seem to introduce biases in the learning processes. Moreover, for some cases, it outperforms the original concept when looking at the first misclassification in the ranking.

Furthermore, we have demonstrated the system performance in a time series case scenario. The system did detect successfully different types of image changes, such as flooded areas or even the crop rotation in agricultural fields. The system is also able to retrieve efficiently unchanged patches.

Finally, we have implemented a link to an external data infrastructure which allows us to include a feature model in the learning processes based on information independent of the image content (e.g., information extracted from GIS maps). This link provides a new tool for the improvement of the AL processes and the posterior search and retrieval operations. Our

tests have shown promising results, allowing the definition of more complex semantic concepts. As future work, we can point out the automation of map model generation, and the introduction of additional models based on heterogeneous sources, (e.g., text metadata). With the inclusion of new models more intensive validation of the obtained results must be performed. For that purpose, we will continue with new benchmarking procedures for larger data sets, with images from different sensors, heterogeneous third party data sources and with a greater scenario variety.

ACKNOWLEDGMENT

The authors would like to thank European Space Imaging (EUSI) for providing the WorldView-2 images as well as the TerraSAR-X Science Service System (Project TELEIOS) for providing the TS-X images. They would also like to thank M. Pinheiro, D. Espinoza-Molina, and G. Schwarz for their help during the writing and reviewing process.

REFERENCES

- [1] K. C. Clarke, "Advances in geographic information systems," *Comput. Environ. Urban Syst.*, vol. 10, no. 3–4, pp. 175–184, 1986.
- [2] R. L. Church, "Geographical information systems and location science," *Comput. Oper. Res.*, vol. 29, pp. 541–562, 2002.
- [3] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, no. 2, pp. 1–60, Apr. 2008.
- [4] C.-R. Shyu, M. Klaric, G. J. Scott, A. S. Barb, C. H. Davis, and K. Palaniappan, "GeoIRIS: Geospatial information retrieval and indexing system—Content mining, semantics modeling, and complex queries," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 4, pp. 839–852, Apr. 2007.
- [5] N. Boujemaa and J. Fauqueur, "Ikona: Interactive specific and generic image retrieval," in *Proc. Int. Workshop Multimedia Content Based Indexing Retrieval*, 2001, pp. 2–5.
- [6] M. Datcu *et al.*, "Information mining in remote sensing image archives: System concepts," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 12, pp. 2923–2936, Dec. 2003.
- [7] K. W. Tobin *et al.*, "Automated feature generation in large-scale geospatial libraries for content-based indexing," *Photogramm. Eng. Remote Sens.*, vol. 72, no. 5, pp. 531–540, May 2006.
- [8] S. Natali *et al.*, "Multitemporal data management and exploitation," in *Proc. Multi-Temp 6th Int. Workshop Anal. Multi-Temporal Remote Sens. Images*, Trento, 2011, pp. 217–220.
- [9] S. Durbha and R. King, "Semantics-enabled framework for knowledge discovery from earth observation data archives," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 11, pp. 2563–2572, Nov. 2005.
- [10] F. Bovolo and L. Bruzzone, "Image information mining in time series: Algorithms and methods for prototyping," Tech. Rep. IIMTS-TN-ACS-0102 Issue 1.0, ESA, 2007.
- [11] M. Molinier, J. Laaksonen, and T. Hame, "Detecting man-made structures and changes in satellite imagery with a content-based information retrieval system built on self-organizing maps," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 4, pp. 861–874, Apr. 2007 [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4137859
- [12] EOlib. (2011, Dec.). *Earth Observation Librarian* [Online]. Available: <http://deepenandlearn.esa.int/tiki-index.php?page=EOlib+Project>
- [13] D. Espinoza-Molina and M. Datcu, "Earth-observation image retrieval based on content, semantics, and metadata," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 11, pp. 5145–5159, Nov. 2013.
- [14] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data—The story so far," *Int. J. Semant. Web Inf. Syst.*, vol. 5, no. 3, pp. 1–22, 2009 [Online]. Available: <http://eprints.soton.ac.uk/271285/>
- [15] M. Quartulli and I. G. Olaizola, "A review of EO image information mining," *ISPRS J. Photogramm. Remote Sens.*, vol. 75, pp. 11–28, Jan. 2013.
- [16] K. Alonso and M. Datcu, "Knowledge-driven image mining system for big earth observation data fusion: GIS maps inclusion in active learning stage," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS'14)*, Quebec City, Canada, 2014, pp. 3538–3541.
- [17] M. Schröder, H. Rehrauer, K. Seidel, and M. Datcu, "Interactive learning and probabilistic retrieval in remote sensing image archives," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 5, pp. 2288–2298, Sep. 2000.
- [18] D. Heckerman, "A tutorial on learning with Bayesian networks," *Microsoft Research Advanced Technology Division*, Tech. Rep. MSR-TR-95-06, Nov. 1996.
- [19] Y. Huang, Z. Wu, L. Wang, and T. Tan, "Feature coding in image classification: A comprehensive study," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, p. 493–506, Mar. 2014.
- [20] J. Chen *et al.*, "WLD: A robust local image descriptor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1705–1720, Sep. 2010.
- [21] M. Haklay and P. Weber, "OpenStreetMap: User-generated street maps," *IEEE Pervasive Comput.*, 2008, pp. 12–18 [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4653466
- [22] QGIS Development Team, *QGIS Geographic Information System*, Open Source Geospatial Foundation, 2014 [Online]. Available: <http://qgis.osgeo.org>
- [23] D. Lewis, "Naive (Bayes) at forty: The independence assumption in information retrieval," in *Machine Learning: ECML-98*, vol. 1398, C. Nédellec and C. Rouveilol, Eds. New York, NY, USA: Springer, 1998, pp. 4–15 [Online]. Available: <http://dx.doi.org/10.1007/BFb0026666>
- [24] M. Schröder-Brzosniowski, "Stochastic modeling of image content in remote sensing image archives," Ph.D. dissertation, ETH - Swiss Federal Institute of Technology, 2000 [Online]. Available: <http://dx.doi.org/10.3929/ethz-a-003851205>
- [25] I. M. G. Muñoz, "Concepts elaboration and system architectures for mining very large image archives," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Univ. of Siegen, 2009 [Online]. Available: <http://www.zess.uni-siegen.de/zebra/reference.php?layout=krb&dd=D2015050811191200056>
- [26] H. Daschiel and M. Datcu, "Information mining in remote sensing image archives: System evaluation," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 1, pp. 188–199, Jan. 2005.
- [27] H. Daschiel and M. Datcu, "Image information mining system evaluation using information-theoretic measures," *EURASIP J. Appl. Signal Process.*, vol. 14, pp. 2153–2163, 2005.
- [28] M. Rehman, M. Iqbal, M. Sharif, and M. Raza, "Content based image retrieval: Survey," *World Appl. Sci. J.*, vol. 19, no. 3, pp. 404–412, 2012.
- [29] A. W. M. Smeulders, M. Worring, S. Santini, 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.
- [30] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi, "The similarity metric," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3250–3264, Dec. 2004.
- [31] P. Grunwald and P. Vitanyi, "Shannon information and kolmogorov complexity," *CoRR*, vol. cs.IT/0410, p. 54, 2004 [Online]. Available: <http://arxiv.org/abs/cs/0410002>
- [32] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006 [Online]. Available: <http://dx.doi.org/10.1016/j.patrec.2005.10.010>



Kevin Alonso received the technical telecommunication engineering degree in speciality of communication systems and the telecommunication engineering degree from the Engineering Technical High School of Bilbao, Bilbao, Spain, in 2005 and 2007, respectively. During 2009 and 2010, he combined his research work on Semantic Web and GIS technologies with the study of a Master's degree in space science and technology. Currently, he is pursuing the Ph.D. degree at German Aerospace Center (DLR), Wessling, Germany.

From July of 2008 to March 2013, he worked with Vicomtech-IK4, San Sebastian, Spain, as a Staff Researcher in the field of Digital Television and Multimedia Services, participating in the European Research Project RUSHES. His research interests include Big Data focused data mining and fusion along with GIS and semantic web technologies and applications for multimedia asset analysis and management.



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

Since 1981, he has been a Professor with the Faculty of Electronics, Telecommunications, and Information Technology, UPB, working in signal/image processing and electronic speckle interferometry. Since 1993, he has been a Scientist with German Aerospace Center (DLR), Wessling, Germany. He is engaged in research related to information theoretical aspects and semantic representations in advanced communication systems. Currently, he is a Senior Scientist and Image Analysis Research Group Leader with the Remote Sensing Technology Institute, DLR, Wessling, Germany. Since 2011, he is also leading the Immersive Visual Information Mining Research

Laboratory, Munich Aerospace Faculty, and is a Director of the Research Center for Spatial Information, UPB. He has held Visiting Professor appointments at the University of Oviedo, Oviedo, Spain, the University Louis Pasteur, Strasbourg, France, the International Space University, Strasbourg, France, the University of Siegen, Siegen, Germany, the University of Camerino, Camerino, Italy, and the Swiss Center for Scientific Computing, Manno, Switzerland. From 1992 to 2002, he had a longer Invited Professor assignment with the Swiss Federal Institute of Technology, ETH Zurich, Zurich, Switzerland. Since 2001, he has initiated and led the Competence Centre on Information Extraction and Image Understanding for Earth Observation, ParisTech, 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 at ParisTech, Telecom Paris. He has authored more than 200 scientific publications, among them about 50 journal papers, and a book on number theory. His research interests include information and complexity theory, stochastic processes, Bayesian inference, and IIM.

Prof. Datcu is a member of the European Big Data from Space Coordination Group.