AN OBJECT-BASED META KNOWLEDGE MODEL FOR A DISTRIBUTED IMAGE INTERPRETATION SYSTEM

G. A. O. P. Costa [a *], P. Hofmann [b], P. N. Happ [c], R. Q. Feitosa [a, c]

[a] Rio de Janeiro State University, Brazil – gilson.costa@ime.uerj.br
[b] University of Salzburg, Austria - peter.hofmann@sbg.ac.at
[c] Pontifical Catholic University of Rio de Janeiro, Brazil – {patrick, raul}@ele.puc-rio.br

KEY WORDS: Remote Sensing, Object-Based Image Analysis, Knowledge-Based Systems, Ontology

ABSTRACT:

This paper introduces the interpretation meta knowledge model devised for the InterCloud platform. InterCloud is a remote sensing image interpretation platform designed to run on computer clusters or on cloud computing infrastructure. The system is capable of distributing data processing tasks, such as segmentation, feature extraction and classification procedures over the processing elements of a computer grid in a transparent way to the user. Moreover, InterCloud can exploit the potential scalability offered by commercial cloud computing infrastructure services, enabling the interpretation of very large remote sensing datasets in an efficient way. The proposed meta model comprises two types of knowledge: declarative and procedural. The former describes the characteristics of the classes of objects expected to be found in the scene to be interpreted, and the relationships among those classes. The latter describes the functions and procedures that should be applied over the data in order to achieve the desired interpretation. In the proposed knowledge model, the user expresses declarative knowledge through the definition of an ontology, so-called descriptive ontology, which conveys the formal naming and definition of the properties and interrelationships of the object classes in a particular application. Procedural knowledge is expressed by the so-called task ontology, which is represented by a directed graph, in which the nodes represent operations over the input images or over the segments generated by segmentation operations. Besides segmentation, crisp or fuzzy classification operations can be defined by the user. The graph edges define the data flow between operations, which are triggered by the control process as soon as their inputs are produced by the preceding operations. In this paper we illustrate the main components of the meta knowledge model through a theoretical application.

1. INTRODUCTION

To this date, many knowledge-based or cognitive systems (Sagerer et al., 1990; Liedtke et al., 1997; Bückner et al., 2001; Costa et al., 2010; Trimble, 2014) have been proposed for remote-sensing image interpretation. The main focus of those systems is the modelling of the classes of objects expected to be found in a scene through the explicit representation of prior knowledge about their spectral, morphological or topological characteristics.

There are several advantages of embodying image understanding knowledge into explicit structures (Crevier and Lapage, 1997). First, knowledge can be easily added to a knowledge base, without modifying pre-existing rules. When laid out explicitly, knowledge can be more easily validated, since contradictions and omissions become apparent. Knowledge structures also favour interactive problem solution, providing a way to explore alternative means of extracting information from images. Last but not least, explicit represented knowledge provides for easier collaboration, for knowledge interchange among those tackling similar problems.

Ontologies represent a way to explicitly specify domain conceptualizations (Gruber, 1993). In the remote sensing domain, ontologies have been used to describe the classes of objects of interest for particular applications, mostly intending to support feature extraction methods (Arvor et al., 2013). For example, Forestier et al. (2012) designed ontologies that represent urban features (e.g. house, garden, roads) and groups of urban features (e.g. urban blocks) in QuickBird imagery.

Forestier et al. (2013) also presented a knowledge-based framework for remote sensing image interpretation of coastal areas. Kohli et al. (2012) developed an ontology dedicated to identifying informal settlements from remotely sensed imagery. Belgiu et al. (2014) developed ontologies of the objects to be identified in remote sensing data by means of machine learning techniques.

In this work, we introduce the meta knowledge model devised for the InterCloud platform. InterCloud is a remote sensing image interpretation platform designed to run on computer clusters. InterCloud can exploit the potential scalability offered by commercial cloud computing infrastructure services, enabling the interpretation of very large remote sensing datasets in an efficient way.

The proposed meta knowledge model comprises two types of knowledge: declarative and procedural. The former describes the characteristics of the classes of objects expected to be found in the scene to be interpreted, and the relationships among those classes. The latter describes the functions and procedures that should be applied over the data in order to produce the desired interpretation.

In the proposed meta model, the user expresses declarative knowledge through the definition of an ontology, so-called descriptive ontology, which conveys the formal naming and definition of the properties and interrelationships of the object classes in a particular application. Such properties can include crisp or fuzzy descriptions of the related features values, which are used by the distributed interpretation control engine in the

direct selection of image objects or in assigning class membership values to the objects.

Procedural knowledge is expressed by the so-called task ontology, which is represented by a directed graph, in which the nodes represent operations over the input images or over the segments generated by segmentation operations. Besides segmentation, crisp or fuzzy classification operations can be defined by the user. The graph edges define the data flow between operations, which are triggered by the control process as soon as their inputs are produced by the preceding operations.

2. INTERCLOUD OVERVIEW

InterCloud is a remote sensing image interpretation platform designed to run on computer clusters or on cloud computing infrastructure. The system employs the MapReduce programing model (Dean et al., 2004) to distribute data processing tasks, such as segmentation (Happ et al., 2015), feature extraction and classification procedures (Ayma et al., 2015) over the processing elements of a computer grid in a transparent way to the user. Moreover, InterCloud can exploit the potential scalability offered by commercial cloud computing infrastructure services, enabling the interpretation of very large remote sensing datasets in an efficient way. InterCloud can be regarded as a redesign of InterIMAGE (Costa et al., 2010) conceived to process arbitrarily large datasets in a distributed fashion.

Although the architecture of InterCloud has been described before (Ferreira et al., 2014), we recall some of its main aspects. The architecture is composed by three abstraction layers: the project definition layer, the image interpretation layer, and the distribution layer.

The project definition layer supports the definition of all required information for the execution of an interpretation application. Such information can be described through semantic networks and dataflow graphs that represent processing chains. The first enables the description of the classes of objects expected to be found by the interpretation process, and the latter enables the description of the chain of operations that should be carried in the interpretation process, and that will be distributed over the processing elements in a transparent way to the end user.

The interpretation layer comprises the algorithms and methods available in the platform. It is structured on a high-level programming language that hides the complexity of dealing directly with the distributed programming model. A user with conventional programming skills can embed new methods into this layer, so that the end user can select them in the project definition layer. Finally, the distribution layer is responsible for the distributed execution of the interpretation model.

In the current implementation of InterCloud, the project definition layer lacks a graphical user interface (GUI), so that end user interaction is restricted by the fact that the user needs to describe the interpretation model through a traditional programming language (i.e., Java). Additionally, InterCloud lacks conceptually structured knowledge representation formalisms to guide the user in the definition of interpretation models, and that provide high level abstractions for the definition of such models, so that application domain experts can define complex models without the support of programmers.

This work intends to fill this gap, by proposing knowledge representation abstractions based on the concept of ontologies. In the following we describe those formalisms, which are currently being considered in the implementation of InterCloud's GUI.

3. META KNOWLEDGE MODEL

In this work we propose an ontology-based meta knowledge model for remote sensing image interpretation. In this section we define the basic components of the meta model and present some examples to illustrate such components in a simple, theoretical application.

Firstly, what we understand by knowledge are facts and heuristics. Facts constitute the body of information available and heuristics are rules of good judgment, of plausible reasoning, that characterize expert-level decision making.

In the case of object-based image interpretation, facts are related to image objects, which represent the occurrence of instances of object classes on the scene represented by an image. Image objects that belongs to a specific class share common characteristics, which are based on particular properties, which are related to features that can be measured or quantified. Such features may be associated to (i) statistics computed over spectral intensity values of the pixels that belong to the image objects, to (ii) to the shape of image objects, to (iii) topological relationships among objects of specific classes, and to (iv) temporal relationships among objects that occur in images acquired at different points in time. In any case, the description of the characteristics shared by objects of a specific class and the relationships among objects from different classes must be formally described in order to be processed by an automated procedure. We call such kind of knowledge *descriptive knowledge*.

In this context, heuristics are associated to the way descriptive knowledge is used or processed, so as to find the location of image objects of the classes of interest, on a particular scene. This is associated to a different kind of knowledge, which has to do with the order of processing of tasks, which aim at measuring the properties of image objects and determining the association of the objects to specific classes. We call such kind of knowledge *procedural knowledge*.

In our meta knowledge model, descriptive knowledge is explicitly represented through a so-called *descriptive ontology*, and procedural knowledge is represented through a *task ontology*.

3.1. Descriptive Ontology

In the proposed meta model, descriptive ontologies are represented by a graphical knowledge representation formalism, which is based on semantic networks (Sagerer and Niemann, 1997).

The nodes of the semantic network may represent either object classes or *lingual descriptors*. Lingual descriptors represent generic characteristics of the objects of a class. Lingual descriptors are defined loosely, in linguistic terms, which are qualitative and imprecise by nature. Such imprecision means

that lingual descriptors cannot be actually computed by an automated process, so they need to be associated to features that can be quantitatively measured. Therefore, each lingual descriptor must be associated to what we call a *descriptor definition*.

A descriptor definition is composed by a set of measurable features, which are associated to corresponding fuzzy sets. Possibility values derived from the fuzzy sets can be combined by fuzzy operators, so that the computation of a descriptor will produce a single possibility value.

As many lingual descriptors may be associated to an object class, lingual descriptors may also be combined through fuzzy operators. In this way, the possibility values associated to each lingual descriptor may be combined so as to produce a single membership value for an image object, which can be later used by a particular process identified in the task ontology, e.g., defuzzification or spatial conflict resolution.

Figure 1 illustrates a simple descriptive ontology. The edges of the semantic network have different semantic meanings. When the edges connect class nodes, they represent structural relationships between classes, such as *is-a* and *part-of*. When the edges connect lingual descriptors they represent operators that combine possibility values, i.e., *t-norms* or *t-conorms*.
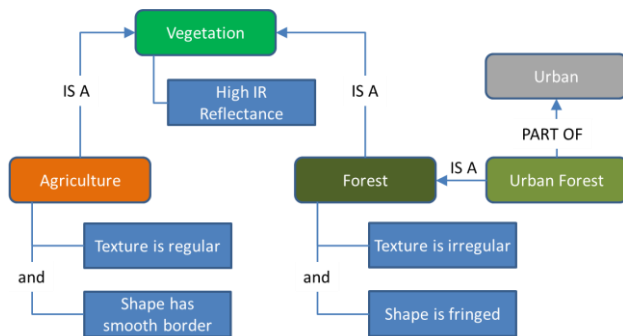

Figure 1. Descriptive ontology example.

Figure 2 shows an example of a lingual descriptor definition. The edges connect feature nodes to the descriptor, they also represent *t-norms* or *t-conorms*.
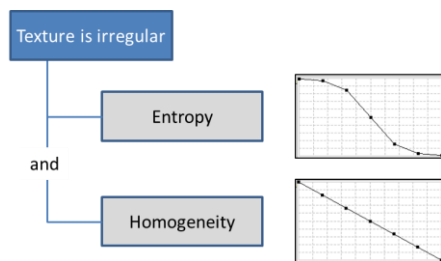

Figure 2. Lingual descriptor definition.

3.2. Task Ontology

The task ontology can be understood as a dataflow description. It represents a processing chain that will somehow use the definitions of the descriptive ontology in the interpretation

process, in order to find occurrences of instances (image objects) of the classes of interest.

Task ontologies are represented by a direct graph, whose nodes represent either processes, data sets (of raster images or vector data), or object classes; and whose edges indicate the processing flow.

Note that, as more than one edge may come out of one node, the edges can be numbered, so that the control process will know the exact order of processing of the connected nodes. Figure 3 shows a task ontology associated to the descriptive ontology illustrated in Figure 1.
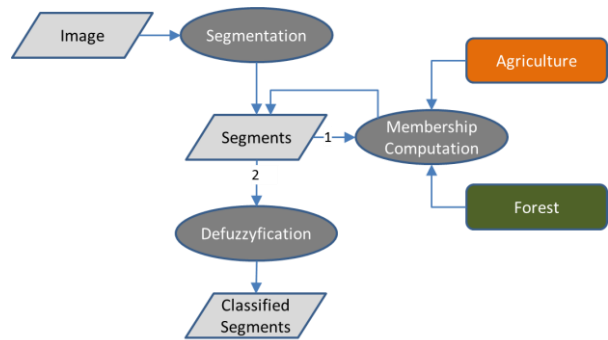

Figure 3. Task ontology example.

It is interesting to observe that for the same descriptive ontology, many different task ontologies may be defined. In Figure 4 we show a different task ontology, associated to the descriptive ontology illustrated in Figure 1. In this task ontology different image segmentations (from different algorithms or from a same algorithm with different parameter values) are performed for the different classes, and a spatial conflict resolution procedure is carried out on the resulting objects, based on their membership values.
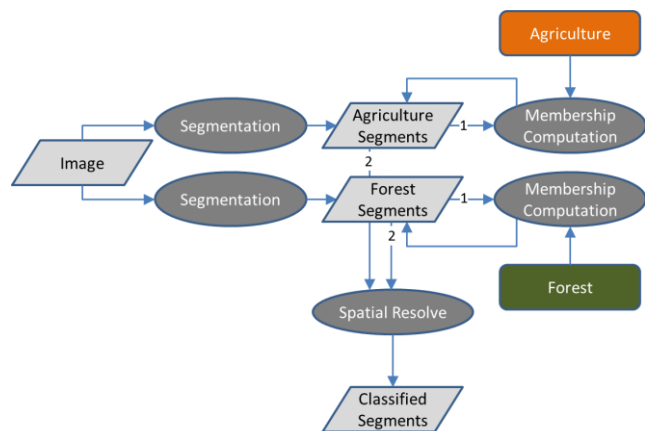

Figure 4. Alternative task ontology.

Figure 5 shows yet another example of a task ontology. This time classification is carried out with a machine learning, supervised technique. In this case, the classification feature space may be defined directly from the features that compose the definition of the lingual descriptors associated to the object classes, or the user may define other features by configuring the parameters of the machine learning procedure.
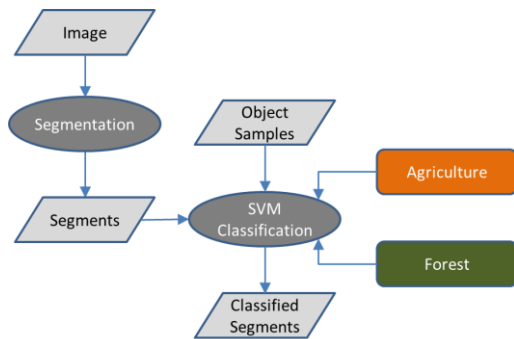
Figure 5. Task ontology with machine learning process.

4. CONCLUSIONS AND FUTURE WORK

In this work we presented a meta knowledge model for InterCloud, a distributed object-based image interpretation system. The meta model is able to represent descriptive knowledge and procedural knowledge, through descriptive ontologies and task ontologies, respectively.

In the descriptive ontologies, the characteristics of object classes are defined though lingual descriptors, which are later specified in terms of descriptor definitions, which associate the linguistic terms to concrete, measurable features computed on the image segments.

We believe that lingual descriptors represent an abstraction level that can facilitate knowledge interchange among users tackling similar problems, thus providing a bridge for transferability of image interpretation models.

The task ontologies enable the definition of complex processing chains, and provide a sensible way to control the image interpretation process.

We are currently in the process of detailing the meta model and the corresponding graphical user interface, which we plan to make available in a future version of InterCloud.

ACKNOWLEDGEMENTS

REFERENCES

Arvor, D., Durieux L., Andrés S., Laporte M.-A., 2013. Advances in Geographic Object-Based Image Analysis with Ontologies: a review of main contributions and limitations from a remote sensing perspective. ISPRS Journal of Photogrammetry and Remote Sensing, 82, 125-137.

Ayma, V.A., Ferreira, R.S., Happ, P.N., Oliveira, D.A.B., Costa, G. A. O. P., Feitosa, R. Q., Plaza, A., Gamba, P., 2015. On the architecture of a big data classification tool based on a map reduce approach for hyperspectral image analysis. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, 2015, pp. 1508-1511.

Belgiu, M., Tomljenovic, I., Lampoltshammer, T., Blaschke, T., Höfle, B., 2014. Ontology-Based Classification of Building Types Detected from Airborne Laser Scanning Data. Remote Sensing, 6, pp. 1347-1366.

Bückner, J., Pahl, M., Stahlhut, O., Liedtke, C.-E., 2001. GEOAIDA - A Knowledge Based Automatic Image Data Analyzer for Remote Sensing Data. In: ICSC Congress On Computational Intelligence Methods and Applications 2001 - CIMA 2001, 2001, Bangor, Wales, Great Britain. Proceedings of the Congress on Computational Intelligence Methods and Applications 2001 - CIMA 2001. International Computing Sciences Conventions - University of Wales in Bangor, U.K.

Costa, G.A.O.P., Feitosa, R.Q., Fonseca, L.M., Oliveira, D.A.B., Ferreira, R.S., Castejon, E., 2010. Knowledge-based interpretation of remote sensing data with the interimage system: major characteristics and recent developments. In: Addink, E., Van Coillie, F. (Eds.), GEOBIA. ISPRS Working Groups, Gent, Belgium, 2010.

Crevier, D., Lepage, R., 1997. Knowledge-Based Image Understanding Systems: A Survey. Computer Vision and Image Understanding, 67 (2), pp. 61-185.

Dean J., Ghemawat, S., 2004. MapReduce: Simplified Data Processing on Large Clusters. OSDI'04: Sixth Symposium on Operating System Design and Implementation, San Francisco, CA, 2004.

Ferreira, R.S., Oliveira, D.A.B., Happ, P., Costa, G.A.O.P, Feitosa, R.Q., Bentes, C., 2014. InterIMAGE 2: The Architecture of an Open Source, High Performance Framework for Automatic, Knowledge-Based Image Interpretation. International Geographic Object-Based Image Analysis Conference, Thessaloniki, 2014.

Forestier, G., Puissant, A., Wemmert, C., Gançarski, P., 2012. Knowledge-based region labeling for remote sensing image interpretation. Computers, Environment and Urban Systems, 36, pp. 470-480.

Forestier, G., Wemmert, C. & Puissant, A., 2013. Coastal image interpretation using background knowledge and semantics. Computers & Geosciences, 54.

Gruber, T. R., 1993. A translation approach to portable ontology specifications. Journal of Knowledge Acquisition for Knowledge-Based Systems, 5, pp. 199-220.

Happ, P.N., Ferreira, R.S., Costa, G.A.O.P., Feitosa, R.Q., Bentes, C., Gamba, P., 2015. Towards distributed region growing image segmentation based on MapReduce. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, 2015, pp. 4352-4355.

Kohli, D., Sliuzas, R., Kerle, N., Stein, A., 2012. An ontology of slums for image-based classification. Computers, Environment and Urban Systems, 36, pp. 154-163.

Liedtke, C.-E., Bückner, J., Grau, O., Growe, S., Tönjes, R., 1999. Aida: A System for the Knowledge Based Interpretation of Remote Sensing Data. In: Third International Airborne Remote Sensing Conference and Exhibition, 1999, Copenhagen, Denmark, Proceedings of the Third International

Airborne Remote Sensing Conference and Exhibition. Environmental Research Institute of Michigan.

Niemann, H., Sagerer, G., Schröder, S., Kummert, F., 1990. ERNEST: A Semantic Network System for Pattern Understanding. IEEE Transactions on Pattern Analysis and Machine Inteligence, 12 (9).

Sagerer, G., Niemann, H., 1997. Semantic Networks for Understanding Scenes: Advances in Computer Vision and Machine Intelligence. 1.ed. New York: Plenum Publishing Corporation.

Trimble, 2014. eCognition Developer 8.7.2. Reference Book, München.