Ontology Evaluation with Protégé using OWLET

Thomas J. Lampoltshammer and Thomas Heistracher

***Abstract*—Amalgamation of formalised knowledge and real-world datasets is a pivotal challenge in the realm of information and communication technologies. Semi-automated classification of datasets can be performed by utilisation of ontologies. The detection process of image objects in Very High Resolution Satellite Imagery (VHRSI) gives a prominent example. The process of refinement of formalised expert knowledge within the related ontology still remains a challenging and time-consuming task. In this paper, the *JSON2OWL Converter (OWLET)* extension for Protégé is presented which supports experts during this refinement phase. The extension offers an integrated approach to transfer real-world dataset objects into the ontology modelling software for semi-automated classification. This transfer is achieved by combining open standard formats from both domains, the (Geo) Web domain (GeoJSON) and the Web ontology domain (OWL2). Thereby *OWLET* supports the process of accuracy analysis and accuracy fostering. By utilising the OWLET extension, experts can not only speed up their classification procedure considerably, but they can also refine their formalised knowledge by using the results of the classification process in conjunction with the outcomes of the accuracy analysis.**

***Index Terms*—OWL, Ontology, Protégé, Remote Sensing, GIS, Knowledge Formalisation**

I. Introduction

As the formalisation of expert knowledge is required in almost all research domains, ontologies represent one solution towards this issue. The challenge associated to this procedure is twofold: on the hand the formalised knowledge has to be applied to the real-world datasets, on the other hand the real- dataset should also enhance and develop the formal knowledge. By serving both sides, semi-automated classification of data becomes possible.

For instance, Object Based Image Analysis (OBIA) presents an accepted and efficient method concerning the classification of high-resolution imagery datasets [1]. The main idea of this approach is to segment the original image into homogeneous units, based on pre-define criteria. In the next step, the image analysis process builds on these segmented objects [2]. Up to now, OBIA is expert knowledge-driven. In consequence, the accuracy of the classification and the time consumption for this process strongly depend on the experts‘ knowledge about the properties of objects represented in the image. Furthermore, he or she has to consider the relations between the objects as well as the context of the image. These circumstances exacerbate the process of embedding OBIA-based methods into operational frameworks, where speed and flexibility of information retrieval are important assets [3], [4]. To overcome this issue, a formalisation of *a priori* knowledge is necessary for the purpose of information extraction from satellite imagery by image analysis systems. One possible solution is presented in form of ontologies. Ontologies can be defined within the Artificial Intelligence (AI) domain as ‘an explicit specification of a shared conceptualisation’ [5, p. 1]. In other words: An ontology can be seen as the knowledge of a domain expert, formalised in a machine-understandable way. Various research endeavours have been conducted to employ ontologies for the automation of the image analysis and interpretation process [6], [7], [8], [9]. For a detailed review of ontology-based applications in remote sensing, please refer to [10]. In summary, it can be argued that the process of image analysis and image interpretation necessitates two kinds of knowledge: i) domain-specific knowledge, and ii) knowledge for image analysis [11]. The domain-specific knowledge represents the domain-specific terminology and incorporated semantics. The image interpretation knowledge can be separated into *qualitative information* and *quantitative information*. The first refers to spectral and spatial properties of objects inherent in e.g. satellite imagery. These properties can be characterised in natural language: for instance, rivers feature an elongated form, or buildings are represented by rectangular objects when seen from above. These qualitative descriptions need then to be mapped to information from the source image, in particular, the delineated objects. One of the most challenging steps presented by the associated engineering process is to form a knowledge base which features the necessary domain-specific semantics at a meaningful granularity. This issues is known in the literature as *‘the ontology grounding problem’* [12]. To contribute towards a possible solution to this problem, the J**SO**N2O**WL** Conve**rt**er (OWLET) presented in this paper aims at supporting domain experts during the development and refinement phase of their ontologies based on real-world data as fundament. This process bridges the ontology domain and the domain of real-world applications.

For the development of before-mentioned ontologies, the ontology creation tool Protégé [13] represents the *de facto* standard in the remote sensing community and beyond. This tool is not an expert system itself. Instead, it is intended to provide the necessary environment to develop custom-tailored tools for the process of knowledge-acquisition. The current version of Protégé is Java-based and therefore platform-independent. Furthermore, it is possible to extend the Protégé environment by plug-ins such as ontology visualisation [14] or fuzzy logics [15], [16]. Therefore, OWLET was realised as such a plugin[1].

Thomas J. Lampoltshammer is with the Department of Geoinformatics - Z_GIS; University of Salzburg; Hellbrunnerstrasse 34; 5020 Salzburg; Austria
Thomas Heistracher and Thomas J. Lampoltshammer are with the School of Information Technology and Systems Management; Salzburg University of Applied Sciences; Urstein Süd 1; 5412 Puch/Salzburg; Austria

[1] A demo version of the plugin, together with sample data and a sample ontology for testing purposes can be found at: http://lampoltshammer.com/owlet/demo.zip

The remainder of the paper is as follows: First, the overall architecture of OWLET is presented, together with the associated data flow and workflow. Second, the transformation process of geo data to the ontology modelling language is described. Subsequently, the presented plugin is demonstrated via an example workflow. A discussion about important aspects of the suggested solution and the conclusion end this paper.

II. Ontology evaluation

Throughout the literature, various approaches exist to evaluate given ontologies. According to [17], four main evaluation directions can be identified: i) 'gold standard' comparison, ii) application-based evaluation, iii) data source comparison, and iv) human-centric evaluation. The first category is dedicated to compare the given ontology to a 'gold standard', that is a predefined, well-formed dataset against other datasets are measured [18]. The second category describes the use of an ontology within an application and then to evaluate the outcome based on the employed ontology [19]. The third category utilises a repository of documents about a certain domain, which is then compared to the ontology that should cover this domain and the associated knowledge [20]. The last category describes a human-centric approach. Here, experts assess the quality of the given ontology by comparing it to a defined set of criteria [21].

The example workflow described in this paper is based on object features from literature, in particular building features. This qualitative and quantitative knowledge is then employed to manually build a 'gold standard', against which the ontology-based classification results are matched.

III. The integrated system architecture and process flow

Figure 1 depicts the integrated system architecture as well as the associated process flow. The rhomboid boxes represent data for input/output, while the rectangular-shaped boxes depict processing modules. The arrows within the figure visualise the process flow within the architecture. The architecture itself comprises several layers from bottom to top of Fig. 1. : i) the data layer, ii) the image processing layer, iii) the reasoning layer, and iv) the expert knowledge layer. The first layer serves as a repository for Very High Resolution Satellite Imagery (VHRSI). From this database, the image to be analysed is handled by the image processing layer. This layer is represented by a remote sensing application for image processing and analysis, e.g. eCognition[2]. Via this object-based image analysis tool, image segmentation algorithms are applied to the satellite image. The process of segmentation can be described as the partitioning of an image into distinct areas or regions. These regions are non-overlapping and are homogeneous considering certain predefined attributes [22]. The resulting delineated objects are then exported into the GeoJSON format [23]. The export functionality is either included in the remote sensing application or can be achieved via the use of external services

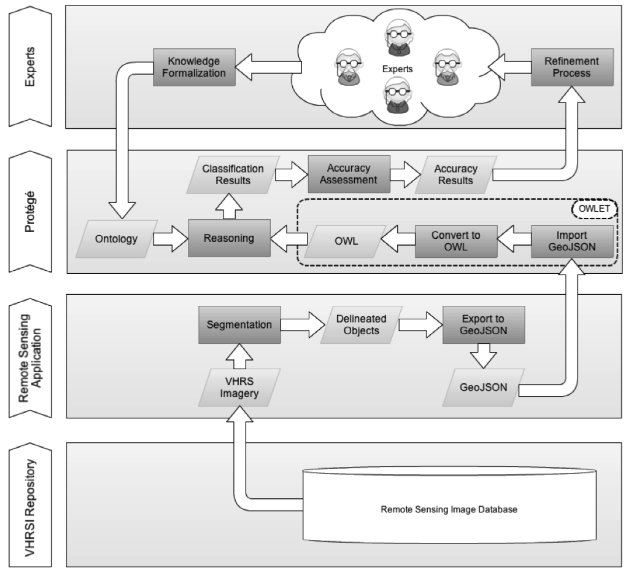

Fig. 1: Integrated architecture of the OWLET plugin

or libraries, such as *WorldMap*[3] or the *Geospatial Data Abstraction Library (GDAL)*[4]. The resulting GeoJSON file is then imported into Protégé via the *OWLET* plugin which is denoted by dashed lines within the processing modules in Fig. 1. In the next step, the imported objects (delineated objects from the prior segmentation) are translated into the Ontology Web Language Version 2 (OWL2) [24]. Subsequently, the ontology and the objects modelled in OWL (so-called Individuals) are now merged for the reasoning process. The ontology itself comprises the qualitative class description of the domain, as well as the quantitative descriptions based on the actual image. The classification results can then be used to perform an accuracy assessment based on *precision* and *recall* [25]. In the final step, experts can analyse characteristics of misclassified objects to refine the qualitative and quantitative knowledge modelled within the ontology, until satisfying accuracy results are achieved.

IV. Geo data transformation

In order to validate and evaluate the developed ontology, the segmented objects from the satellite image have to be imported into Protégé. As Protégé does not 'understand' shape files, the segmented objects have to be exported and converted in order to be compatible. In the first step, the given shape file has to be exported into the GeoJSON format [23]. GeoJSON provides encoding capabilities for various geographic data structures, such as geometry, a feature, or a collection of features. After the export is completed, the newly produced GeoJSON file can be imported into Protégé via *OWLET*.

For the process of reasoning (in this case the process of classification based on formalised knowledge), Protégé employs description logics in form of various reasoners. These reasoners make use of the specifically designed elements of

[2] eCognition - http://www.ecognition.com

[3] WorldMap - http://worldmap.harvard.edu/

[4] GDAL - http:www.gdal.org/

the Ontology Web Language Version 2 (OWL) by W3C [24]. For a comprehensive treatise of the origin of OWL2, the reader may refer to [26]. The OWL vocabulary contains three main artefacts as there are classes, individuals and, properties. Classes as such can be described as set of individuals, while properties describe the relationships between the classes and in consequence the associated individuals. These artefacts are formalised within OWL based on Description Logics (DL) – here called OWL DL. By these logic statements, automated testing by a reasoner becomes possible. A reasoner can be described as application the can infer logical relationships within the ontology and in consequence can perform consistency, equivalence and instantiation testing. There exist three profiles within the OWL2 standard - also called fragment or sublanguage - available. These sublanguages represent altered versions of OWL, namely: i) OWL 2 EL, ii) OWL 2 QL, and iii) OWL 2 RL (Motik et al., 2009). For the ontology in this research work, the author chose Protégé-OWL [27], an adapted and optimised version for Protégé.

This part of the paper introduces the main components utilised to build a basic ontology in OWL. First, a class hierarchy is built by classes and sub-classes. Listing 1 describes the class ‘Area’ to be a sub-class of the parent class ‘Classifiers’.

```
<Declaration>
        <Class IRI="#Area"/>
</Declaration>
<Declaration>
        <Class IRI="#Classifiers"/>
</Declaration>
<SubClassOf>
        <Class IRI="#Area"/>
        <Class IRI="#Classifiers"/>
</SubClassOf>
```

Listing 1: Defining OWL classes and sub-classes

In order to link two classes with each other, object properties are employed. In addition, these relationships can be employed to describe entire classes - called ‘EquivalentClasses’. Listing 2 demonstrates how such properties and ‘EquivalentClasses’ are described in OWL. In particular, it is defined that a class ‘ResidentialAres’ is equivalent to a class that is linked to the class ’LowArea’ via the object property ‘hasArea’.

```
<EquivalentClasses>
  <Class IRI="#ResidentialArea"/>
  <ObjectIntersectionOf>
    <ObjectSomeValuesFrom>
         <ObjectProperty IRI="#hasArea"/>
          <Class IRI="#LowArea"/>
    </ObjectSomeValuesFrom>
  </ObjectIntersectionOf>
</EquivalentClasses>
```

Listing 2: Defining OWL EquivalenClasses by ObjectProperties and Relationships

The next step consists of mapping these qualitative descriptions to quantitative values. Again, the principle of ‘EquivalentClasses’ is used. In addition, data properties are utilised to map concrete values to qualitative descriptions (see List. 3). Here, the definition of ’LowArea’ is described as a double value of greater than 1,500. The property which holds this value is denoted as ‘area_pxl’.

```
<EquivalentClasses>
      <Class IRI="#LowArea"/>
      <DataSomeValuesFrom>
          <DataProperty IRI="#area_pxl"/>
          <DatatypeRestriction>
              <Datatype abbreviatedIRI="xsd:double"/>
              <FacetRestriction facet="&xsd;maxExclusive">
                   Literal datatypeIRI="&xsd;double
                       ">1500.0</Literal>
              </FacetRestriction>
          </DatatypeRestriction>
      </DataSomeValuesFrom>
</EquivalentClasses>
```

Listing 3: Mapping of qualitative and quantitative knowledge

What the OWLET plugin does is to parse the objects denoted in the GeoJSON file and ports them into the OWL syntax to be included into the ontology. Listing 4 shows an example ontology entry for a parsed object as an individual.

```
<Declaration>
    <NamedIndividual IRI="#DataSet_building_set.2"/>
</Declaration>

<DataPropertyAssertion>
    <DataProperty IRI="#area_pxl"/>
    <NamedIndividual IRI="#DataSet_building_set.2"/>
    <Literal datatypeIRI="&xsd;double">1230.0</Literal>
</DataPropertyAssertion>
```

Listing 4: Defining OWL classes and sub-classes

This process is repeated for all properties of one object and for all objects included within the GeoJSON file.

V. Application Example

The following example describes a typical application scenario for OWLET. The example at hand is simplified for demonstration purposes. The employed quantitative and qualitative descriptions within the ontology are by no means representative. However, the described workflow can easily be extended to cover complex knowledge acquisition projects as well. For a more complex example dedicated to classification of buildings from Light detection and ranging (LiDAR)-based data, involving an early prototypical version of this plugin, the reader may refer to [28]. As a start, a remote sensing image is envisioned (for instance VHRSI) after the segmentation process in eCognition. The desired task is to identify different types of building classes within the segmented objects. In the next step, the segmented objects are exported to the GeoJSON format. The features utilised in this example are building features from literature. In particular, features mentioned in the work of [29] are employed such as i) the occupied area of the building, ii) the density of buildings in a specific area, iii) the two-dimensional shape of the building, and iv) the roof type of the building. An example set of segmented objects after the export to GeoJSON can be seen in Fig. 2.

The ‘manual_classification’ field holds the associated manual (visual) classification by the expert. Type ‘1’ represents an ‘ResidentialArea’ and Type ‘2’ represents an ‘IndustrialArea’. The generated GeoJSON file can then be imported by the OWLET plugin. The user interface of the extension as such is clean and simple. Via a file explorer, the user can select the specific GeoJSON file which is then imported into the ontology.

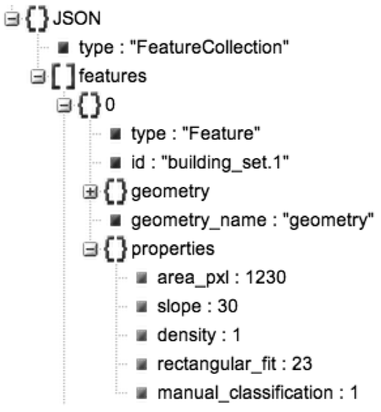

Fig. 2: GeoJSON example file after export

It is important to use the same object properties from the segmented objects within the ontology. A visualisation of the modelled *a priori* knowledge is depicted in Fig. 3. This figure represents a snapshot of the larger ontology, which was embedded in the *Corine* land-use/land-cover ontology[5]. After the import of the segmented objects within the GeoJSON file, the objects are contained as individuals within the ontology. In the next step, the reasoning process can be started to perform the ontology-based classification.

Based on the combined results per class (Tab. I and Tab. II), *precision* (1) and *recall* (2) can be calculated. In this case, the precision and recall values reach 100% and 70% respectively for the class 'ResidentialArea', while the associated precision and recall values for the "IndustrialArea' class reach 80% for precision and 50% for recall.

	Condition pos.	Condition neg.
Test pos.	392	0
Test neg.	168	240

TABLE I: Precision and Recall for 'ResidentialArea' class

	Condition pos.	Condition neg.
Test pos.	120	30
Test neg.	120	530

TABLE II: Precision and Recall for 'IndustrialArea' class

$$Precision = \frac{true\,positive}{true\,positive + false\,positive} \tag{1}$$

$$Recall = \frac{true\,positive}{true\,positive + false\,negative} \tag{2}$$

This situation occurred due to the 'unsharp' properties of some objects. An example object can be seen in Fig. 4. This object was manually classified as 'IndustrialArea', but was neither recognised as 'ResidentialArea' nor 'IndustrialArea'. Some properties are very close to both types of classes and therefore it is not possible to come up with a distinct result.

[5] Corine - http://harmonisa.uni-klu.ac.at/de

At this point, it is the expert's task to decide upon the quantitative description of the classes and their refinement. In addition, the expert might need to add additional qualitative descriptions as well. Furthermore, it could turn out that the existing object properties are not enough to describe the given classes in a proper way and additional properties have to be included or existing properties have to be omitted due to their generality.

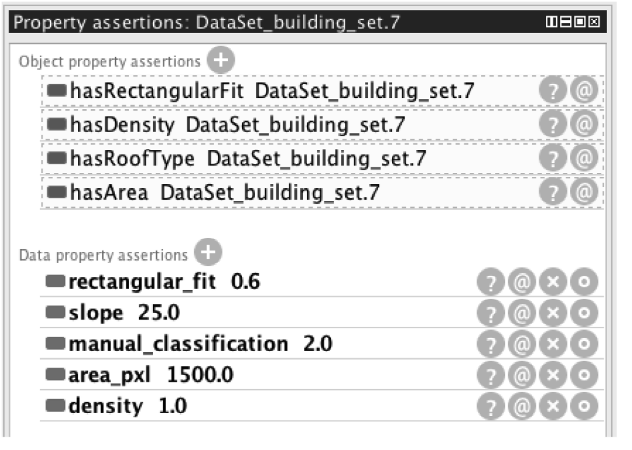

Fig. 4: Misclassified object

VI. Discussion and Conclusion

As Protégé is the *de facto* software in regard to ontology development, the introduced extension does not only provide a higher degree of comfort, but also speeds up the entire evaluation process. As manually defining hundreds of segmented objects and their associated attributes would demand a tremendous amount of time, the import and translation capabilities of the OWLET plugin perform these tasks in a few minutes or even a few seconds - depending on the amount of objects.

However, several potential pitfalls remain, which should not be neglected. Three issues are related to the work with ontologies and quantitative modelling, while the fourth issue is an inherent problem of the remote sensing domain. The first issue related to the work with ontologies is represented by the so-called *semantic gap* [30]. This issue describes the fact that visual description of data is biased by the analysts' own perception and experiences. Hence, assuming n experts work on the interpretation task, n different interpretations may result. However, this issue of 'an objective reality' is not novel and was (and still is) discussed in philosophy under the paradigm of 'constructivism' [31].

Another issue can be identified as the problem of 'overfitting' [32]. This issues occurs if a model comprises more details than necessary (such as attributes or terms) to define a given concept. This can lead to worse decisions during the classification process, as the 'tweaked' ontology may cover some types of object classes better, while others are now misclassified. In addition, overfitting has a severe impact on the ontology's transferability.

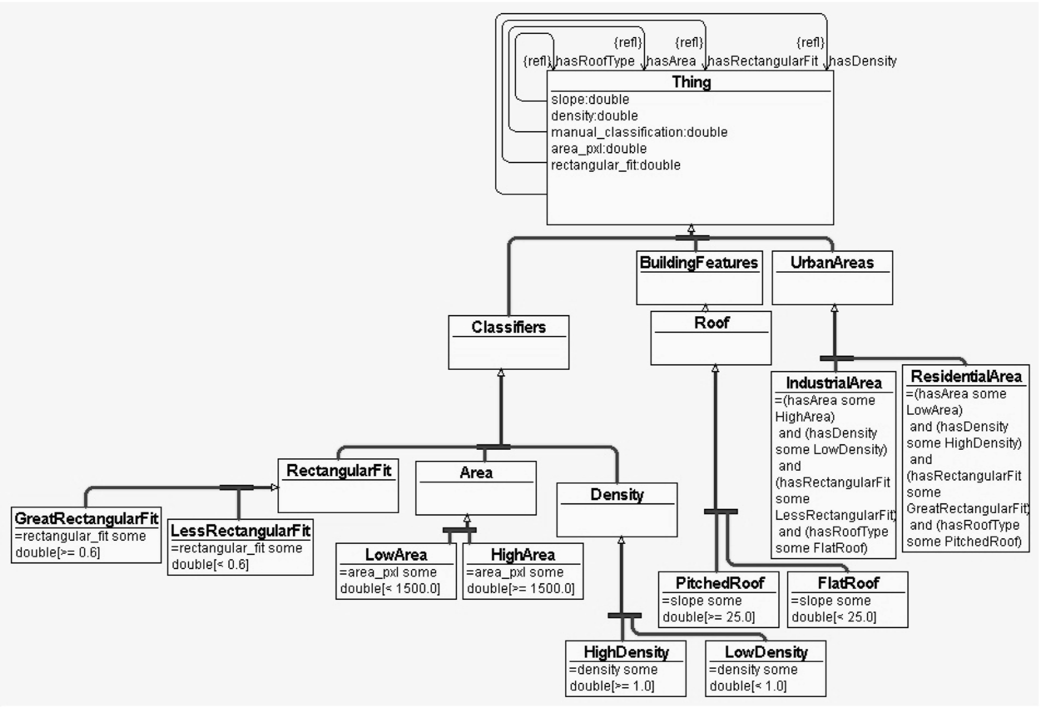

Fig. 3: Experts‘ a priori knowledge formalised within an OWL ontology in Protégé

For instance, an expert refines the ontology’s quantitative and qualitative descriptors to match the given image close to 100% of the ‘gold standard’. If the actual image has some special image properties, these are modelled within the ontology as well. If the expert now tries to transfer the newly gained ontology to another image from the same domain – e.g. another snapshot of a different area of the same city – these additional modelled properties have now the potential to negatively influence the classification results.

When working with ontology reasoning, the topic performance is an important aspect. Performance issues related to reasoning in terms of complexity or computational resources for various reasoners were studied by the authors of [33] and [34].

There come several reasoners included in Protégé ‘from the shelf’. One of them is called *Pellet* and comprises state-of-the-art optimisation techniques such as Normalisation, Simplification, Absorption and Semantic Branching [35]. In addition, it features novel approaches to improve performance when handling nominal (enumerated classes) and individuals. As the approach in this paper strongly deals with individuals, Pellet poses as one solution for the reasoning process. Still, the user has to bear in mind that large numbers of individuals (several thousands) may heavily impact on the overall performance in terms of computation times and memory resources.

Figure 5 depicts the overall classification performance of the three common reasoners for Protégé. It can be seen that if the number of objects increases, the classification time is rapidly rising around 1,000 objects. This behaviour can become mission critical when dealing with near-real-time applications.

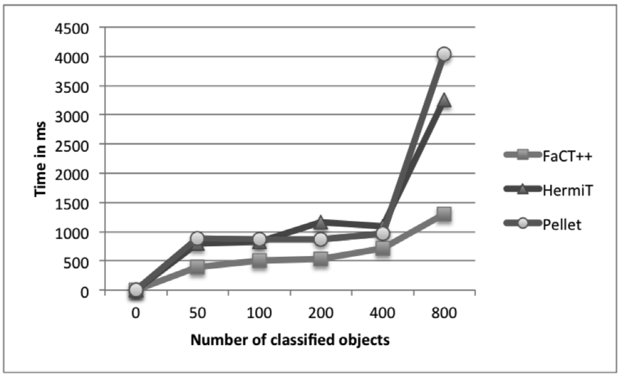

Fig. 5: Performance and run time comparison of the main reasoners of Protégé. The x-axis represents the number of objects classified, while the y-axis shows the associated classification time in ms.

The last issue to be discussed is an inherent problem of the remote sensing domain itself – the segmentation process [36]. Image segmentation refers to the process of aggregating adjacent pixels based on their similarities such as texture. As the related algorithms for the segmentation process tend to react heavily on small changes to the segmentation parameters, the outcomes of the segmentation may vary significantly. Users of the presented plugin may therefore not neglect that this important step in the process flow will impact the overall accuracy of the ontology – even if the ontology itself would be ‘100%’ accurate.

This paper presented the OWLET plugin for Protégé to deliver an integrated solution for the process of ontology evaluation. This open-source extension supports domain experts from numerous field during the iterative process of formalising his or her knowledge, while performing collateral evaluation and refinement. The introduced integrated methodology for semi-automated classification of (image) objects, together with the implemented plugin serves as the missing link in the defined process. In addition, the suggested process, as well as its implementation, relies on multiple international standard technologies and tools such as JSON, OWL2 and Protégé. For the classification task, we map extracted (image) object information against a formalised a priori expert knowledge in form of an ontology. Furthermore, we demonstrate how to refine the modelled knowledge based on the classification outcomes. Our approach is built on state-of-the-art technologies; it is open and generic and can thus be adopted by people from various research fields.

ACKNOWLEDGMENT

This research is funded by the Austrian Science Fund (FWF) and the Salzburg University of Applied Sciences through the Doctoral College GIScience (DK W 1237-N23).

REFERENCES

[1] T. Blaschke, "Object based image analysis for remote sensing," *ISPRS journal of photogrammetry and remote sensing*, vol. 65, no. 1, pp. 2–16, 2010.

[2] P. Hofmann, J. Strobl, and A. Nazarkulova, "Mapping green spaces in bishkek—how reliable can spatial analysis be?" *Remote Sensing*, vol. 3, no. 6, pp. 1088–1103, 2011.

[3] N. S. Anders, A. C. Seijmonsbergen, and W. Bouten, "Segmentation optimization and stratified object-based analysis for semi-automated geomorphological mapping," *Remote Sensing of Environment*, vol. 115, no. 12, pp. 2976–2985, 2011.

[4] L. Moller-Jensen, "Classification of urban land cover based on expert systems, object models and texture," *Computers, environment and urban systems*, vol. 21, no. 3, pp. 291–302, 1997.

[5] T. R. Gruber *et al.*, "A translation approach to portable ontology specifications," *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[6] G. Forestier, A. Puissant, C. Wemmert, and P. Gançarski, "Knowledge-based region labeling for remote sensing image interpretation," *Computers, Environment and Urban Systems*, vol. 36, no. 5, pp. 470–480, 2012.

[7] F. de Bertrand de Beuvron, S. Marc-Zwecker, A. Puissant, and C. Zanni-Merk, "From expert knowledge to formal ontologies for semantic interpretation of the urban environment from satellite images," *International Journal of Knowledge-Based and Intelligent Engineering Systems*, vol. 17, no. 1, pp. 55–65, 2013.

[8] M. Thonnat, "Knowledge-based techniques for image processing and for image understanding," *Journal de Physique 4*, vol. 12, no. 1, pp. Pr1–189, 2002.

[9] P. Hofmann, P. Lettmayer, T. Blaschke, M. Belgiu, S. Wegenkittl, R. Graf, T. J. Lampoltshammer, and V. Andrejchenko, "Abia – a conceptional framework for agent based image analysis," *South-Eastern European Journal of Earth Observation and Geomatics*, vol. 3, no. 2s, pp. 125–130, 2014.

[10] D. Arvor, L. Durieux, S. Andrés, and M.-A. Laporte, "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*, vol. 82, pp. 125–137, 2013.

[11] C. Hudelot and M. Thonnat, "A cognitive vision platform for automatic recognition of natural complex objects," in *Tools with Artificial Intelligence, 2003. Proceedings. 15th IEEE International Conference on*. IEEE, 2003, pp. 398–405.

[12] W. Kuhn, "Ontologies in support of activities in geographical space," *International Journal of Geographical Information Science*, vol. 15, no. 7, pp. 613–631, 2001.

[13] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubézy, H. Eriksson, N. F. Noy, and S. W. Tu, "The evolution of protégé: an environment for knowledge-based systems development," *International Journal of Human-computer studies*, vol. 58, no. 1, pp. 89–123, 2003.

[14] A. Katifori, E. Torou, C. Halatsis, G. Lepouras, and C. Vassilakis, "A comparative study of four ontology visualization techniques in protege: Experiment setup and preliminary results," in *Information Visualization, 2006. IV 2006. Tenth International Conference on*. IEEE, 2006, pp. 417–423.

[15] S. Calegari and D. Ciucci, "Fuzzy ontology, fuzzy description logics and fuzzy-owl," in *Applications of Fuzzy Sets Theory*. Springer, 2007, pp. 118–126.

[16] M. Belgiu, T. Lampoltshammer, and B. Hofer, "An extension of an ontology-based land cover designation approach for fuzzy rules," in *GI_Forum 2013. Creating the GISociety*, A. Car, T. Jekel, and J. Strobl, Eds. Vienna: Austrian Academy of Sciences Press, 2013, pp. 59–70.

[17] J. Brank, M. Grobelnik, and D. Mladenić, "A survey of ontology evaluation techniques," in *In In Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD 2005*. Citeseer, 2005.

[18] A. Maedche and S. Staab, "Measuring similarity between ontologies," in *Knowledge engineering and knowledge management: Ontologies and the semantic web*. Springer, 2002, pp. 251–263.

[19] R. Porzel and R. Malaka, "A task-based approach for ontology evaluation," in *ECAI Workshop on Ontology Learning and Population, Valencia, Spain*. Citeseer, 2004.

[20] C. Brewster, H. Alani, S. Dasmahapatra, and Y. Wilks, "Data driven ontology evaluation," in *International Conference on Language Resources and Evaluation (LREC 2004), 24-30 May 2004, Lisbon, Portugal.*, 2004.

[21] A. Lozano-Tello and A. Gómez-Pérez, "Ontometric: A method to choose the appropriate ontology," *Journal of Database Management*, vol. 2, no. 15, pp. 1–18, 2004.

[22] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation 1," *Annual review of biomedical engineering*, vol. 2, no. 1, pp. 315–337, 2000.

[23] H. Butler, M. Daly, A. Doyle, S. Gillies, T. Schaub, and C. Schmidt, "The geojson format specification," 2008.

[24] P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, and S. Rudolph, "Owl 2 web ontology language primer," *W3C recommendation*, vol. 27, pp. 1–123, 2009.

[25] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval.* ACM press New York, 1999, vol. 463.

[26] I. Horrocks, P. F. Patel-Schneider, and F. Van Harmelen, "From shiq and rdf to owl: The making of a web ontology language," *Web semantics: science, services and agents on the World Wide Web*, vol. 1, no. 1, pp. 7–26, 2003.

[27] H. Knublauch, R. W. Fergerson, N. F. Noy, and M. A. Musen, "The protégé owl plugin: An open development environment for semantic web applications," in *The Semantic Web–ISWC 2004*. Springer, 2004, pp. 229–243.

[28] M. Belgiu, I. Tomljenovic, T. J. Lampoltshammer, T. Blaschke, and B. Höfle, "Ontology-based classification of building types detected from airborne laser scanning data," *Remote Sensing*, vol. 6, no. 2, pp. 1347–1366, 2014. [Online]. Available: http://www.mdpi.com/2072-4292/6/2/1347

[29] G. Sohn and I. Dowman, "Extraction of buildings from high resolution satellite data," *Automated Extraction of Man-Made Objects from Aerial and Space Images (III). Balkema Publishers, Lisse*, pp. 345–355, 2001.

[30] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 12, pp. 1349–1380, 2000.

[31] D. H. Jonassen, "Objectivism versus constructivism: Do we need a new philosophical paradigm?" *Educational technology research and development*, vol. 39, no. 3, pp. 5–14, 1991.

[32] D. M. Hawkins, "The problem of overfitting," *Journal of chemical information and computer sciences*, vol. 44, no. 1, pp. 1–12, 2004.

[33] J. Bock, P. Haase, Q. Ji, and R. Volz, "Benchmarking owl reasoners," in *Proc. of the ARea2008 Workshop, Tenerife, Spain (June 2008)*, 2008.

[34] Y. Li, Y. Yu, and J. Heflin, "Evaluating reasoners under realistic semantic web conditions," in *Proceedings of the 2012 OWL Reasoner Evaluation Workshop*, 2012.

[35] F. Baader, *The description logic handbook: theory, implementation, and applications*. Cambridge university press, 2003.

[36] G. Meinel and M. Neubert, "A comparison of segmentation programs for high resolution remote sensing data," *International Archives of Photogrammetry and Remote Sensing*, vol. 35, no. Part B, pp. 1097–1105, 2004.