



**HAL**  
open science

## From EO Change Rasters to Knowledge Graphs: An approach Based on Regions of Interest

Jordane Dorne, Nathalie Aussenac-Gilles, Catherine Comparot, Romain Hugues, Cassia Trojahn

### ► To cite this version:

Jordane Dorne, Nathalie Aussenac-Gilles, Catherine Comparot, Romain Hugues, Cassia Trojahn. From EO Change Rasters to Knowledge Graphs: An approach Based on Regions of Interest. 4th International Workshop on Geospatial Linked Data (GeoLD 2021@ESWC 2021), Beyza Yaman; Mohamed Ahmed Sherif; Axel-Cyrille Ngonga Ngomo; Armin Haller, Jun 2021, Hersonissos, Greece. pp.67-79. hal-03327705

**HAL Id: hal-03327705**

<https://hal.science/hal-03327705v1>

Submitted on 27 Aug 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# From EO Change Rasters to Knowledge Graphs: An approach Based on Regions of Interest

Jordane Dorne<sup>1,2</sup>, Nathalie Aussenac-Gilles<sup>1</sup>, Catherine Comparot<sup>1</sup>, Romain  
Hugues<sup>2</sup>, Cassia Trojahn<sup>1</sup>

<sup>1</sup> Institut de Recherche en Informatique de Toulouse, France

`firstname.lastname@irit.fr`

<sup>2</sup> Thales Alenia Space, France

`firstname.lastname@thalesaleniaspace.com`

**Abstract.** This paper proposes a process that supports the full life-cycle to generate and exploit knowledge graphs (KG) from Earth observation rasters and open data. The innovative features of this process include i) an algorithm to automatically identify Regions of Interest (ROIs) on a raster, offering an accurate geographic division, i.e. geolocated references used to guide data integration; ii) a semantic-driven process to generate a KG from different sources (raster, open, linked and social data); and iii) a validation of the approach with a use case on fire detection, which shows the added value of KGs to identify ROIs where high changes have been detected.

**Keywords:** satellite images, change detection, knowledge graphs.

## 1 Introduction

Thanks to efficient deep learning models, Earth Observation satellite images (EO images) can be more easily compared so that changes between series of two or more images can be computed. Changes are indicators that may reveal natural events or disasters such as fires, floods, or earthquakes; changes in vegetation like frost impact on cultures, or damaged forests after hurricanes; or long term evolution of ground occupation like deforestation, urbanisation, etc. Services assessing geographic changes over time typically require searching, discovering, and manually identifying relevant data. This is a difficult and laborious task that relies on domain knowledge and that is often subject to human error [21]. Automatically identifying changes thanks to EO satellite image processing reduces this effort. Whereas most early works relied on supervised ML algorithms [20], more recent ones are unsupervised to avoid manual tagging of examples before training their models [3]. These automatic change detection methods compare series or pairs of images in which they identify changes to be represented at pixel level in raster files (we refer to ChAI, for Change Area Index). However, results of this process lack context and reading raster files requires specific tools. Without more information, a detected change cannot help domain experts analyzing those images, understanding the phenomenon, and taking the appropriate decision.

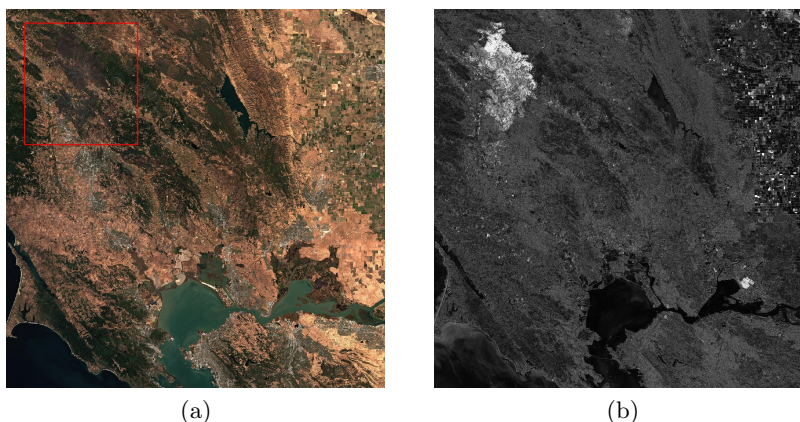


Fig. 1: EO images of California area – (a) Sentinel-2 image taken on 2019/11/01 (red rectangle indicates an area affected by a fire); (b) Change raster computed from image (a) and a previous one (taken on 2019/10/22). Black and white pixels represent respectively low and high probabilities of change.

As an example, let's consider Figures 1(a) and 1(b), which depict snapshots of the same area affected by an important fire, that can hardly be detected by human eye when comparing the two images. Automatic change detection algorithms are much more efficient, as seen in Figure 1(b) where change areas appear in white. Generic change algorithms are efficient to detect changes, to assign them a degree of importance (high, medium, low) but they do not identify the nature of the event that caused the change. To do so, more contextual data is needed. Various kinds of geospatial data can provide context to detect changes and highlight specific kinds of phenomenon: either indices (e.g., vegetation, hydrography or landscape indices) directly computed from EO images; or linked data and social data used as rich complementary sources of information. Providing contextual knowledge to explain or annotate "what can be seen" or "what can be interpreted" on EO images has been the goal of many projects for the past 15 years. This process requires the *integration of data* from various sources and of different nature, which can be efficiently done thanks to semantic web technologies.

Unlike most of previous works that deal with a single type of event (i.e. flood [16] or wild fires [17]), we wanted to design a generic process so that (i) given the type of events to be identified from the change impact of EO images, any kind of dataset could be selected and ingested as semantic contextual knowledge, (ii) it could take as input any change raster whatever the algorithm that generated it; (iii) it could manage the geometry of events with a shape that could reach a compromise between precision (to know the event exact location) and simplicity (to make it easy to compare with the geometry of other geographic objects). The task we address here can be formulated as such: given the results of an unsupervised change detection algorithm on an area of interest, we want to

build a knowledge graph (KG) depicting various contextual data related to each event of a specific kind identifiable on that region at the considered period. One difficulty we address here is to optimally group pixels as for defining relevant regions, i.e., polygons representing Regions Of Interest (ROIs) and to find the accurate geographic division that will guide data integration. Similar to the notion of tiling grid defined by ESA, that divides the Earth surface into tiles representing a fixed area on this surface, the notion of ROI is convenient to be used as a geolocated reference for data integration. Unlike tiles, not all ROIs have the same size or shape. The innovative features of the proposed process include a) an algorithm to identify ROIs on a raster; b) a semantic-driven process and a vocabulary to generate a KG from different sources in order to help explaining the changes identified in change rasters; iii) a validation of the approach with use-cases on fire detection, which shows the added value of KGs to explain automatically ROIs where high changes have been detected.

In the following, Section 2 overviews the approach and details the KG generation process. Section 3 describes its evaluation in terms of change explanation applied to a use-case. Section 4 discusses the main related works. Finally, Section 5 concludes the paper and presents future work.

## 2 Representing ROIs in a Knowledge Graph

The process that we propose takes as input a raster file providing geospatial data, i.e. the value of a given parameter for each pixel in the raster. This raster maybe either a change raster or an index raster. We define these notions in the next section. The goal of the process is to compute ROIs from this raster by aggregating "close" pixels for which the parameter has a minimum value. The process outputs an RDF KG representing these ROIs according to a formal vocabulary. The overall process can be described in 4 steps presented in Section 2.3) and detailed in the following sections.

### 2.1 Input data: Change rasters and index rasters

We consider two kinds of rasters generated from EO images: index rasters, computed from a single image and change rasters, obtained by comparing two images. An *index raster* results from an image analysis at the pixel level. Several dedicated algorithms are available according to the phenomenon to be identified, like nature of the land cover, probability of fire or water flood. For each image pixel, they compute some of its spectral values to generate a numeral value for this index. For instance, the algorithm that computes the Burned Area Index (BAI) produces a raster with values between -1 and 6. Several index rasters are available as open data like Corine Land Cover. A *change raster* (Figure 1 (b)) is produced by a change detection algorithm (as further developed in Section 3) from a couple of satellite images having the same footprint, and captured at different dates. The footprint and the dates are part of the raster metadata. Its content is a grid in which each pixel is associated a change value according to the change level computed by the algorithm. In our use cases, changes are identified by an unsupervised machine learning algorithm. Each pixel has a value between 0 (low probability of change) and 1 (high probability of change).

## 2.2 Regions of Interest (ROI)

In image processing a *Region Of Interest* (ROI) is a subset of an image identified for a particular purpose. In the case of EO images, ROIs give some information about an area on Earth (the footprint). Thus an ROI can be geolocated thanks to a polygon included in the raster area. ROIs can be automatically identified thanks to the raster file analysis. In this case, criteria on pixel values determine what is an area of interest. In this paper, ROIs are not calculated from an image raster, but rather from a raster file that provides some information at pixel level for the same area and at the same date. Such raster files are the above-mentioned change rasters or index rasters with a link towards the image raster(s) that they are computed from. More precisely, we define an ROI as a subset of a raster file selected according to two criteria : i) most of the pixels in this subset have a value higher than a minimum threshold (which reveals something to be noticed); and ii) the polygon of adjacent high level pixels has a minimum size. To speed up the computation and the search for ROI, we decided to represent them by rectangles corresponding to their bounding box. Practically, the ROI is *the bounding box including several close rectangles that gather pixels from a raster file with a value higher than a threshold value*. Each region includes spots of interest rather than delimiting their exact perimeter.

## 2.3 Overview of the KG generation process

The KG generation process takes a change raster and the various contextual datasets as input, and generates a graph of RDF triples. The notion of ROI plays a major role in this process as a means to aggregate information available at the pixel level inside continuous larger regions. This notion is computed in a bottom-up way for each parameter or index using the values of this parameter for each pixel. The process starts with the selection of the kind of event to be studied, and the knowledge required to characterize this type of event. A list of indices and several sets of contextual data are selected for their relevance. For instance, in the validation use case which focuses on wild fires (Section 3), we selected the following types of information as relevant to identify and locate a fire: indices like Change Area Index (ChAI) and Burned Area Index (BAI), location names and tweets as a way to confirm that a fire happened. We get this information from two kinds of data sources: the EO images themselves, from which indices are calculated thanks to specific algorithms, and open data sources. We give here an overview of the main steps of the process. We explain how ROIs are computed and used for triplication in the next sections.

- 0. Raster selection** We select a raster file according to the information it conveys: either change data or index data. Among the raster metadata, we will exploit the date, footprint and the links towards one or 2 images from which it was generated.
- 1. Identifying regions of interest** Given a raster file, we extract the image footprint (its geometry) and the raster period. Then, parsing the raster, we look for ROIs thanks to two parameters: i) the minimum threshold to

consider a pixel value; and ii) the minimum area of a polygon of adjacent high level pixels.

2. **Retrieving contextual data** Considering the footprint geometry and the change period, contextual data with spatial and temporal properties is retrieved from different data sources and in various formats (shapefile, WKT, CSV, ...): open data, open Linked Data, (i.e. or private data).
3. **Generating the KG** A set of transformation rules is applied to extract the appropriate data from each selected source and to create an RDF graph. These RDF graphs are conceptually linked together into one KG, as soon as a node identifier (URI) appears in each of them.

The resulting KG can be used to retrieve the number of ROIs affected by a phenomenon and their size. In a wider scope, the KG can also be used to tackle more specific downstream analysis tasks: visualization of feature changes over time, together with the corresponding satellite images; exploration of supplementary information related to the ROI using additional knowledge bases that we link to the graph for further extending the analysis indicators; etc.

## 2.4 ROI generation

To generate ROIs, we developed Algorithm 2.4, which takes a raster file as input, and produces a shapefile containing all polygons with highest values, and then computes their envelopes as bounding-boxes.

**Polygonizing change areas.** From a change raster our algorithm produces first (line 1) a new raster with binary values (0 and 1) where the value 1 characterizes a pixel having a change value above a given threshold. A polygonal representations of each change area is then computed (line 2).

**Computing ROI envelopes.** From this list of polygons, those with an area below a threshold (provided by *minArea*) are first excluded (lines 3 - 8). A step of simplification and aggregation of close polygons is repeated as long as the list of polygons cannot be reduced any further by number. In fact each execution of the loop (lines 10 - 14) simplifies the list further. The function *Envelope()* (line 12) simplifies the shape of each polygon given their geometry and returns the minimum bounding box of each of them. The function *CascadedUnion()* (line 13) joins polygons that intersect and returns their bounding boxes (their geometry). At the end of the process all resulting polygons are disjoint and have an area superior to the minimum area (the *minArea* parameter) defined to consider a polygon as a ROI. Figure 2 shows the stages and results of the simplification and aggregation process applied on the top left part of the raster of the Figure 1(b) after 6 steps: (a) represents the output of the Polygonize function, i.e. a shapefile that contains thousands of small disjoint polygons; (b) to (g) are the results of each iteration that takes the envelope of each polygon and joins the ones that overlap; (g) is the final result where no more simplification can be applied.

## 2.5 Generating a knowledge graph

To integrate this data, like [23], we rely on the spatial dimension of entities. to reduce the cost of computing spatial relationships between entities representing

---

**Algorithm 1** The ROI generation from a raster

---

```
1:  $highIndexRaster \leftarrow FilterHighIndex(raster, threshold)$ 
2:  $listPolygons \leftarrow Polygonize(highIndexRaster)$ 
3:  $listROIs \leftarrow \emptyset$ 
4: for each  $polygon$  in  $listPolygons$  do
5:   if  $polygon.area \geq minArea$  then
6:      $listROIs.add(polygon)$ 
7:   end if
8: end for
9:  $nbPolygons \leftarrow 0$ 
10: while  $len(listROIs) \neq nbPolygons$  do
11:    $nbPolygons \leftarrow len(listROIs)$ 
12:    $listROIs \leftarrow Envelope(listROIs)$ 
13:    $listROIs \leftarrow CascadedUnion(listROIs)$ 
14: end while
15: return  $listROIs$ 
```

---

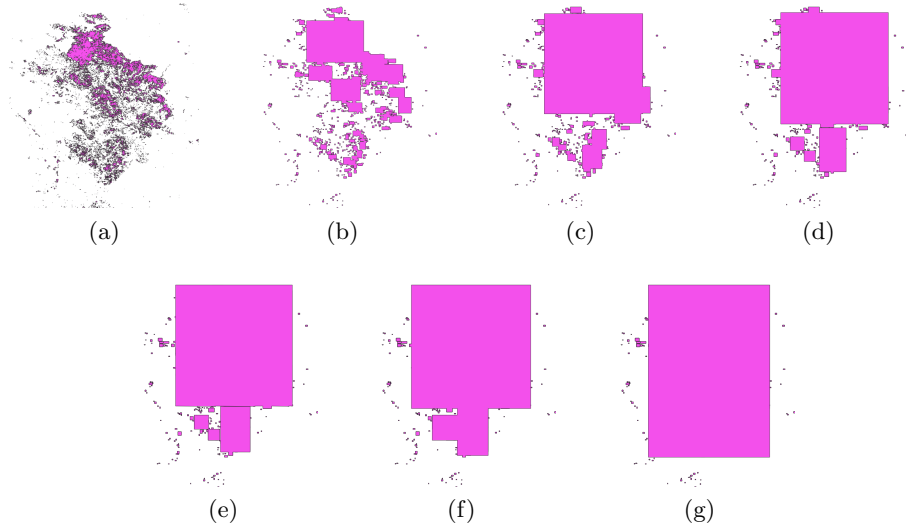


Fig. 2: Evolution of ROIs per iteration during the process of simplification.

data and images, in the sense of [22]. However, we chose to limit ourselves to the spatial relationships defined by GeoSPARQL in order to use this language to query the data. In a second step, the integration takes into account the temporal and spatial properties of the data, as detailed in the following. The KG generation (Figure 3) is based on a vocabulary we have defined reusing standard vocabularies, namely GeoSPARQL [15] (for geospatial data) and OWL-Time [12] (for temporal series), with respectively the *geo* and *time* prefixes. Change raster footprints, ROIs, BAIs are instances of *geo:Feature*; they are described by a geometry (a polygon). ChAIs and BAIs are linked to instances of

*time:TemporalEntity* using the *time:hasTime* property. The temporal dimension of a ROI is provided by a property that links the ROI to its raster. The KG generation is similar for all data sources using a specific vocabulary for each of them. Once data has been extracted from a dataset according to spatial and temporal features, it is converted to JSON format. Then, a Python script uses a template to generate RDF triples from the JSON file. A *template* defines the mapping between a JSON schema and an RDF model to be instantiated. For each data source, we wrote one specific mapping template (in Turtle format) adapted from previous work [2]. These templates and functions help to perform sophisticated mapping operations that are not possible in alternative approaches like RML rules. Moreover, they make it easy to enrich the KG with new data, or to generate a new graph if either the data source schema or the modelling vocabulary are changed.

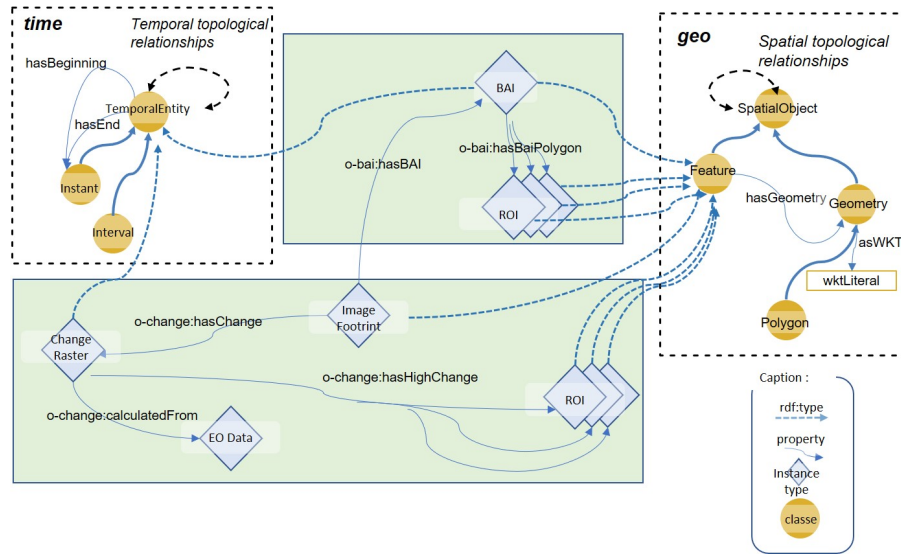


Fig. 3: Extract of the vocabulary used to describe the mapping templates.

### 3 Evaluation

We evaluate the quality of ROI generation and RDF representation in the scope of fire detection from EO images. Because computing the change raster is still quite time consuming, we start with a qualitative evaluation on two images, which provide two use-cases. The first one is the Kincade Fire (11/2019, Figure 1), whereas the second one is the Camp Fire introduced in [6]. The experiment targeted two evaluations: i) the relevance of the generated ROIs; ii) the ability to link the ROIs with contextual data in an RDF graph and to query this graph.

**Data sources** For each use case, we collect several raster files and select data sets from open sources: i) *two EO raster files* of the same area, from the Sentinel-2 collection, captured before and after a fire took place; ii) *a ChAI raster file*



(Figure 1(b)) produced by Thales Alenia Space from these two images thanks to an unsupervised machine learning algorithm [4]; iii) *the BAI raster file* computed on the most recent image using the equation provided by [9]; iv) *reference fire points*, i.e. from FireCast<sup>3</sup>, we collect only the firepoints located on the image area having a confidence index greater than 0.65; v) *administrative units* with a geometry (a polygon) in WKT format that intersects the raster footprint geometry querying the SPARQL endpoint Yago2geo<sup>4</sup>; v) *keywords* extracted from tweets<sup>5</sup> that contain the name of administrative units and may help identifying the nature of changes. From the text in these tweets, we chose to keep the 5 keywords with the highest frequency.

**Generated KG** A KG is generated from ROIs as described in 2.5. It is also enhanced using the other data sources. Administrative units and fire points are instances of *geo:Feature*. The spatial dimension of a tweet keyword is the geometry of the administrative unit associated with it. Tweet keywords and fire points are linked to *time:TemporalEntity* instances.

**Setting Change ROI thresholds** To validate the Change ROI generation algorithm (Algorithm 2.4), we set up optimal values for two thresholds. Both of them are used to reduce the number of generated ROIs on a raster. The first parameter, the *threshold pixel value*, is used by the "polygonize" algorithm as the minimum value for a pixel to be selected in a polygon. It is used to generate a mask that filters out pixels that form polygons. The second parameter to be configured is the *minimum area of a polygon* in square meters. It plays the role of a second filter to select, among all the polygons proposed by polygonize, the one to be considered as a relevant ROI according to the type of event to be studied. The smaller the size, the larger the number of output polygons, many of them being not relevant. On the contrary, choosing a too high value may lead to silence: less polygons are generated, so few that some important ones may be missing. We empirically observed that a pixel value threshold below 0.65 led to noise whereas a threshold higher than 0.65 generates too few polygons, leading us to set this parameter to 0.65. Then we can choose a minimum polygon size between 5,000  $m^2$  and 15,000  $m^2$  to generate a reasonable amount of polygons. We selected 10,000  $m^2$  as minimum area value of a polygon because 1 hectare is a reference unit in remote sensing and especially for the study of forest fires.

**Evaluation of Change ROI generation.** To evaluate the use-case output polygons we use a shapefile provided by the California government. This shape file represents the exact perimeter of the Kincade fire. We aim at demonstrating that the calculated change ROI covers most of the fire surface. In Figure 4(a) we can see that almost the entire perimeter of the Kincade fire is contained in the main ROI polygon generated by our algorithm. According to the CalFire website, the Kincade fire area was 31468ha and the generated ROI that covers this shape is 70905ha. Thanks to the area we are able to identify where the most

<sup>3</sup> <https://firecast.conservation.org/>

<sup>4</sup> <http://yago2geo.di.uoa.gr/>

<sup>5</sup> Tweets are extracted from a collection captured at IIRIT since 2015 and stored on the OSIRIM platform <https://osirim.irit.fr/site/fr/articles/corpus>

important change is located. ROIs allows us to interpret a change without seeing the satellite image nor the change raster.

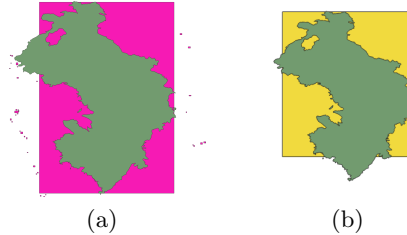


Fig. 4: Kincadee fire (green) over (a) change ROI (pink) - (b) BAI ROI (yellow).

**Evaluation on BAI ROI generation.** This evaluation is similar to the change ROI. Generating BAI polygons aims at extracting from the image all highly burnt areas, i.e. the areas having a high burn index. To do so, we set the threshold value of the burn index to 1 (on a scale from -1 to 6). Each polygon gives us the location of a previous fire thanks to its bounding box. The polygonize function produces much less polygons on BAI rasters than on change rasters. However we applied the same threshold value for the minimum polygon area:  $10,000 m^2$  (1ha). As we can see in Figure 4(b) that most of the Kincadee fire perimeter is included in the generated BAI polygon.

**Evaluation of the data query.** The query available online<sup>6</sup> retrieves all the ROIs, the burnt area in the ROI, the number of linked fire points (from FireCast) and the keywords about the cities in the ROI for the period. From the result of this query, we can get the ROI that contains the highest number of fire points in the area of this raster and the keywords associated to the administrative unit that intersects with this ROI. Keywords can contribute to determine if there is a fire in the ROI. In Figure 5 (a), we can see that a ROI intersects with two administrative units. The word “fire” is one of the keywords detected in tweets at this period for the city of Healdsburg in California. Hence, contextual data explains that the change observed on this area was caused by a fire. Data from tweets properly documents the kind of event that caused the detected change.

**Evaluation on a second use case.** For the Camp Fire which took place in California in November 2018<sup>7</sup> the authors detect the forest fire (a burn map) using change detection over a CNN-based framework without contextual data. Using the same parameters as the one defined for the first use case, we obtained similar results. As shown in Figure 5 (b) we successfully found keywords “fire” and “wildfire” for the concerned administrative units. This use-case enabled us to use Landsat 8 images as input for the change detection algorithm. Even if the spatial resolution is lower we retrieved the same quantity of data.

**Enriching results from unsupervised learning.** The RDF triples can also contribute to validate the results from the machine learning algorithm. To do

<sup>6</sup> <https://bit.ly/2KFKf2p>

<sup>7</sup> <https://www.fire.ca.gov/incidents/2018/11/8/camp-fire/>

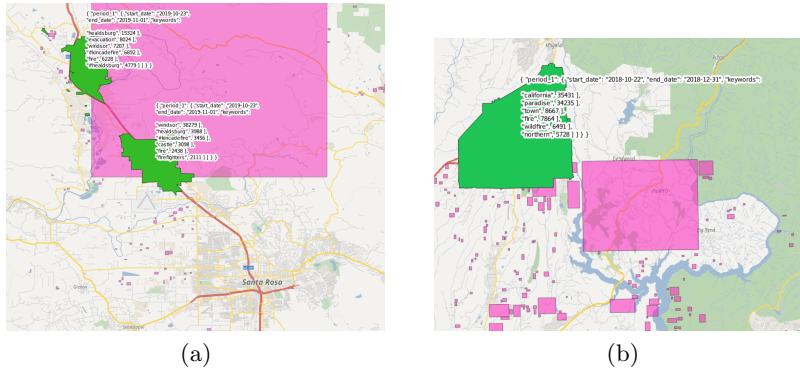


Fig. 5: Keywords for administrative units perimeters (green) and change ROIs (pink) for (a) the Kincade Fire - (b) the Camp Fire.

so, we cross-checked the results obtained from the different sources on the same location and period of time as in the previous section. The results from deep learning process used for change detection show a high change on the image. The Firecast data also show many fire points in the area that explain that the change is due to a fire. This information is also corroborated by the data collected from tweets, that report a forest fire called Kincade Fire <sup>8</sup> in this area of California during this period. Contrariwise, one can observe a fire in a smaller area in the lower right part of the image, as indicated by a grouping of fire points. However, this fire has not been detected as a significant change from the change raster because there are not enough pixels with a value above 0.66 to detect an abnormal change. Nevertheless, our processing chain was able to detect multiple fire points thanks to the Firecast data (Figure 4). In that way, our approach is able to identify false negative areas of the change. Moreover, it corroborated the changes detected by the unsupervised algorithm. In a further work, RDF data could be more tightly coupled with the change learning process to improve its results, for instance by providing feedback or training examples.

## 4 Related work

Annotating the images with external data or with image indices has been addressed in different studies. In [18], vegetation indices result from satellite image processing and are exposed as RDF triples using GeoSPARQL [15]. Close to that approach, in [14] statistical data is collected from open sources to monitor the deforestation of Amazonian rainforest and translated into RDF. In [17], background information (OpenStreetMap, GeoNames, etc.) is used to identify hot spots concerned by a wildfire, in order to refine the process and to reject hot spot candidates. The work made by [11] uses heuristics to detect changes between two satellite images, extracting contextual change information such as damage caused by natural hazards. With a learning perspective, [1] applies a Convolution Neural Network (CNN) to detect different classes of land cover in

<sup>8</sup> <https://www.fire.ca.gov/incidents/2019/10/23/kincade-fire/>

the satellite image (vegetation, railroad, etc.), which are enriched with semantic labels from OpenStreetMap and government sources. In [13] the proposal is to inserting semantic meaning into detected change areas with hypermaps. They apply their CNN to the TSUNAMI Panoramic Change Detection dataset and re-annotate the changed areas with semantic classes (car, building and rubble). In this line, [8] proposes a CNN-based learning approach that simultaneously performs change detection and land cover mapping, while using the predicted land cover information to better predict changes. Also close to our study, [19] enriches change detection over Sentinel-1-A images with event detected in media content (news, posts). In terms of geo-spatial and temporal data linking, interlinking data on EO means discovering spatial and temporal links among RDF graphs [7]. It is close to what we have done in terms of data integration. To compute links between Linked Open Data resources with temporal, and thus event-like properties, the work in [10] uses the intervals of Allen’s algebra, optimally implementing them to more quickly perform the temporal property comparisons needed to compute temporal relationships. Another way of linking is referred to entity matching (or *entity resolution*). It is a matter of associating equivalent entities, for which general semantic link discovery (*entity linking*) can be reused [5,23]. According to [23], most approaches search for equivalence between entities (same labels, names or types), leaving other types of relationships, e.g. spatial or temporal relations, unexploited. However, the spatial representation of most geo-located data is complex, taken the form of a polygon. A technique of *blocking* consists of dividing the earth’s surface into curved rectangles and then evaluating the topological relations between entities based on this division. Similarly, [22] propose to discover topological links even more efficiently by indexing entities using tiles cutting the land surface into rectangles.

## 5 Conclusions

This paper proposed a process for generating a KG for the semantic qualification of automatically computed changes between two images. This process is guided by the notion of Regions of Interest (ROI), as an accurate geographic division representing a high probability of changes. This can be a basic brick in a more complex process of monitoring events on the Earth’s surface, as it is able to automatically identify the location, nature, beginning and end of events. We plan to carry out new experiments to confirm the generality of the integration process, using similar or new types of events (i.e. urbanization, migration flows, ...) and other types of contextual data. We will also investigate how to reuse the RDF data to improve the change detection process. Another task is to improve the keyword selection using a Tf.Idf algorithm to get more relevant keywords and to validate these words against a lexical field of natural disasters. A second perspective would be the use of trained neural models as additional knowledge sources when extracting keywords.

*Acknowledgements* This work benefited from CANDELA H2020 grant (convention number 776193); and ANRT CIFRE PhD. scholar (number 2017/1399).

## References

1. M. Alirezaie, A. Kiselev, and e. a. Långkvist. An ontology-based reasoning framework for querying satellite images for disaster monitoring. *Sensors*, 17, 11 2017.
2. H. Arenas, N. Aussenac-Gilles, C. Comparot, and C. Trojahn. Semantic integration of geospatial data from earth observations. In *EKAW*, pages 97–100, 2016.
3. A. Asokan and J. Anitha. Change detection techniques for remote sensing applications: a survey. *Earth Science Informatics*, 12(2):143–160, Jun 2019.
4. M. Aubrun, A. Troya-Galvis, M. Albughdadi, R. Hugues, and M. Spigai. Unsupervised learning of robust representations for change detection on sentinel-2 earth observation images. In *13th Int. Symp. on Environmental Software Systems*, 2020.
5. S. Auer, J. Lehmann, and A.-C. Ngonga Ngomo. *Introduction to Linked Data and Its Lifecycle on the Web*, pages 1–75. 2011.
6. Y. Ban, P. Zhang, and e. a. Nascetti. Near Real-Time Wildfire Progression Monitoring with Sentinel-1 Time Series and Deep Learning. *SR*, 10, 2020.
7. L. M. V. Blázquez, V. Saquicela, and Ó. Corcho. Interlinking geospatial information in the web of data. In *AGILE Conference*, pages 119–139, 2012.
8. R. C. Daudt, B. L. Saux, A. Boulch, and Y. Gousseau. High resolution semantic change detection. *CoRR*, abs/1810.08452, 2018.
9. F. Filippini. Bais2: Burned area index for sentinel-2. In *Proc. of the 2nd International Electronic Conference on Remote Sensing*, volume 2 (7), page 7, 2018.
10. K. Georgala, M. A. Sherif, and A. N. Ngomo. An efficient approach for the generation of allen relations. In *ECAI*, pages 948–956, 2016.
11. S. Hashimoto, T. Tadono, and M. O. et al. A framework of ontology-based knowledge information processing for change detection in remote sensing data. In *IEEE International Geoscience and Remote Sensing*, pages 3927–3930, 2011.
12. J. R. Hobbs and F. Pan. An ontology of time for the semantic web. *ACM Transactions on Asian Language Information Processing*, 3:66–85, 2004.
13. H. Kataoka, S. Shirakabe, Y. Miyashita, A. Nakamura, K. Iwata, and Y. Satoh. Semantic change detection with hypermaps. *CoRR*, abs/1604.07513, 2016.
14. T. Kauppinen, G. Mira de Espindola, J. Jones, A. Sanchez, B. Graeler, and T. Bartoschek. Linked brazilian amazon rainforest data. *Semantic Web J.*, 5, 01 2013.
15. D. Kolas, M. Perry, and J. Herring. Getting started with GeoSPARQL. Technical report, OGC, 2013.
16. K. R. Kurte and S. S. Durbha. Spatio-temporal ontology for change analysis of flood affected areas using remote sensing images. In *FOIS Workshop*, 2016.
17. K. Kyzirakos, M. Karpathiotakis, and G. et. al. Wildfire monitoring using satellite images, ontologies and linked geospatial data. *Web Semantics*, 24, 01 2014.
18. A. J. Luque, R. Pérez, F. J. Bonet, and P. Magaña. An ontological system based on modis images to assess ecosystem functioning of natura 2000. *Journal Applied Earth Observation and Geoinformation*, 37:142–151, 05 2015.
19. N. Pittaras, G. Papadakis, and e. a. Stamoulis. Geosensor: Semantifying change and event detection over big data. In *SAC*, pages 2259–2266, 2019.
20. R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. *IEEE Trans. Image Proc.*, 14(3):294–307, 2005.
21. B. Shbita, C. A. Knoblock, W. Duan, and et. al. Building linked spatio-temporal data from vectorized historical maps. In *Semantic Web*, pages 409–426, 2020.
22. M. A. Sherif, K. Dreßler, P. Smeros, and A. N. Ngomo. Radon - rapid discovery of topological relations. In *AAAI*, pages 175–181, 2017.
23. P. Smeros and M. Koubarakis. Discovering spatial and temporal links among RDF data. In *Workshop on Linked Data on the Web*, 2016.