remote sensing

MDPI

Article

A KG-Based Integrated UAV Approach for Engineering Semantic Trajectories in the Cultural Heritage Documentation Domain

Konstantinos Kotis [1,*], Sotiris Angelis [1], Efthymia Moraitou [1], Vasilis Kopsachilis [2], Ermioni-Eirini Papadopoulou [2], Nikolaos Soulakellis [2] and Michail Vaitis [2]

[1] Intelligent Systems Lab, Department of Cultural Technology and Communication, University of the Aegean, 81100 Mytilene, Greece

[2] Cartography and Geoinformation Lab, Department of Geography, University of the Aegean, 81100 Mytilene, Greece

* Correspondence: kotis@aegean.gr; Tel.: +30-6974822712

Abstract: Data recordings of the movement of vehicles can be enriched with heterogeneous and multimodal data beyond latitude, longitude, and timestamp and enhanced with complementary segmentations, constituting a semantic trajectory. Semantic Web (SW) technologies have been extensively used for the semantic integration of heterogeneous and multimodal movement-related data, and for the effective modeling of semantic trajectories, in several domains. In this paper, we present an integrated solution for the engineering of cultural heritage semantic trajectories generated from unmanned aerial vehicles (UAVs) and represented as knowledge graphs (KGs). Particularly, this work is motivated by, and evaluated based on, the application domain of UAV missions for documenting regions/points of cultural heritage interest. In this context, this research work extends our previous work on UAV semantic trajectories, contributing (a) an updated methodology for the engineering of semantic trajectories as KGs (STaKG), (b) an implemented toolset for the management of KG-based semantic trajectories, (c) a refined ontology for the representation of knowledge related to UAV semantic trajectories and to cultural heritage documentation, and (d) the application and evaluation of the proposed methodology, the developed toolset, and the ontology within the domain of UAV-based cultural heritage documentation. The evaluation of the integrated UAV solution was achieved by exploiting real datasets collected during three UAV missions to document sites of cultural interest in Lesvos, Greece, i.e., the UNESCO-protected petrified forest of Lesvos Petrified Forest/Geopark, the village of Vrissa, and University Hill.

Keywords: geoinformatics; UAV; knowledge graph; semantic trajectory; ontology; cultural heritage; monitoring; documentation

check for updates

Citation: Kotis, K.; Angelis, S.; Moraitou, E.; Kopsachilis, V.; Papadopoulou, E.-E.; Soulakellis, N.; Vaitis, M. A KG-Based Integrated UAV Approach for Engineering Semantic Trajectories in the Cultural Heritage Documentation Domain. *Remote Sens.* **2023**, *15*, 821. https://doi.org/10.3390/rs15030821

Academic Editors: Andreas Georgopoulos and Dimitrios Skarlatos

Received: 10 December 2022
Revised: 28 January 2023
Accepted: 30 January 2023
Published: 31 January 2023

1. Introduction

Managing large volumes of heterogenous spatiotemporal data is vital for emerging research and development areas, such as unmanned aerial vehicle (UAV) missions for documenting regions/points of cultural heritage interest. The next generation of spatiotemporal knowledge, represented as knowledge graphs (KGs), will integrate multiple and multimodal datasets that contain interlinked geospatial and temporal references, e.g., movement data, weather data, and points and regions of interest (POIs/ROIs). Linked spatiotemporal data (LSD) provide the data infrastructure for geospatial information systems in building the next generation of spatiotemporal data-driven applications, e.g., autonomous vehicles for parcel delivery, for security and surveillance, and for aerial-based documentation. This paper presents an approach to managing LSD to facilitate the engineering of cultural heritage semantic trajectories, and especially for building the next-generation spatiotemporal applications in the domain of documentation UAVs.

The segments of an object's movement, which have been defined based on the interest that they present for a specific application, e.g., UAV movement within a specific area for a given recording/documentation mission, are called trajectories of the moving object [1]. A trajectory can be (a) enriched with additional data (beyond latitude, longitude, timestamp), and/or (b) enhanced with complementary segmentations, constituting a semantic trajectory [2]. Annotations for the segmented parts of a trajectory (episodes) could be "stop" or "move", or in other cases recordings of POIs or ROIs. In terms of deployment, a semantic trajectory may be useful for applications that require the interpretation of the trajectory of a moving object, e.g., POIs/ROIs that a UAV has documented with appropriate recordings (photos/videos). Furthermore, selected knowledge discovery tasks can be performed in semantic trajectories to extract patterns based on characteristics such as changes in movement, POIs/ROIs, or specific behaviors that can be recognized in a single or a group of trajectories, used to classify or recommend future trajectories. An example semantic trajectory depicting a documentation behavior at the Geopark is depicted in Figure 1.

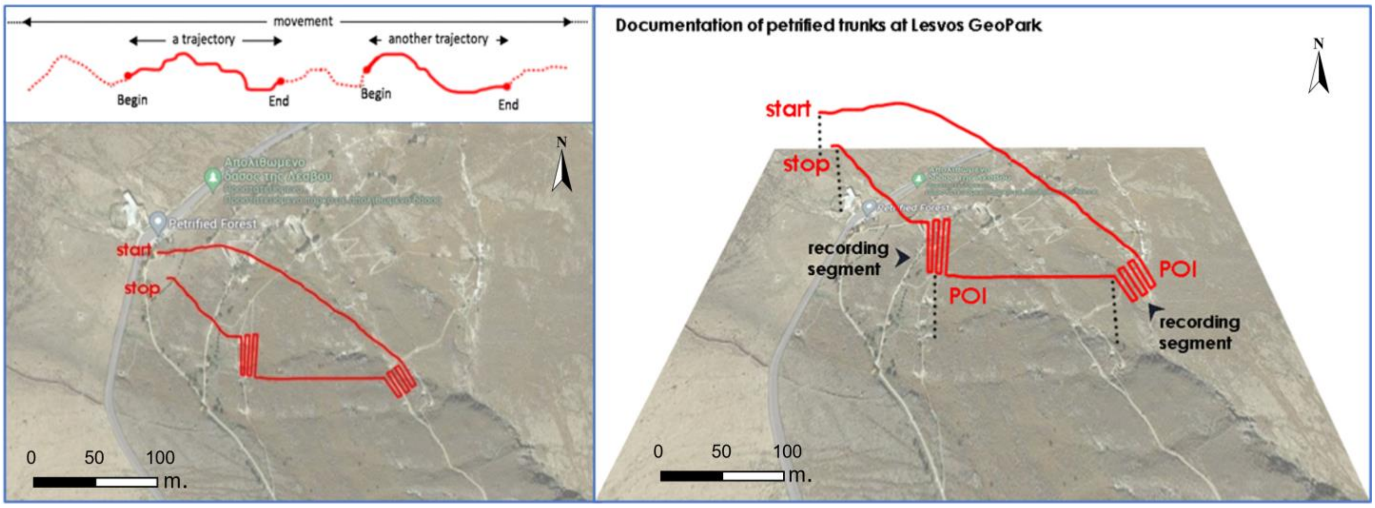


Figure 1. Example semantic trajectory depicting a documentation behavior on the Geopark, starting from a simple trajectory, and resulting in a semantic trajectory.

A semantic trajectory of a swarm of drones is a synthesis of the semantic trajectories of multiple units moving (flying) in a specified formation, sharing common origin–destination points, having a common mission, enriched with semantic annotations at different levels of detail, and having one or more complementary segmentations, where each segmentation consists of a list of annotated episodes. A UAV (drone) trajectory is a sequence of points (traces) that specify the position of the moving entity in space and time. A segment is a part of the trajectory that contains a list of episodes. Each episode has a starting and ending timestamp, segmentation criterion (annotation type), and episode annotation. For example, an annotation type could be the "weather conditions" and an episode annotation could be "a storm", "heavy rain", "extremely high waves", etc.

Semantic Web (SW) technologies have been utilized for the modeling and enrichment of semantic trajectories, since they facilitate (a) the modeling and interlinking of data that can enhance a trajectory of raw movement data, and (b) the segmentation of the trajectories themselves, based on semantics, in a standardized and meaningful way [2,3]. Knowledge graphs (KGs) incorporate semantic models utilized for the structured and formal representation of heterogeneous and multimodal data, as well as for reasoning with multiple integrated views of it [4,5].

Motivated by real-use cases of UAVs' missions to document specific regions and points of cultural heritage interest, such as the documentation of the petrified forest on Lesvos Island, the goal is to develop an integrated tool-supported approach for transforming the trajectories of UAVs into semantic trajectories that can be effectively managed, visualized, and analyzed. A snapshot of such a use case is depicted in Figure 2, in the form of a simplified KG. The objectives of the presented approach are (a) the modeling of semantic trajectories of UAVs, their flights and recordings per mission (e.g., volume and frequency of recording episodes), (b) the visualization and analysis of UAVs' semantic trajectories, (c) the retrieval of semantic information of flights/missions (e.g., UAV positions, recorded POIs, episodes' date/time, weather data), and, (d) the retrieval of records (e.g., photos) that have been produced during different recording events of trajectories related to a flight/mission, based on parameters such as the type or location of recording events (e.g., nearby recording positions, photograph recording, the object of interest that has been recorded, etc.).

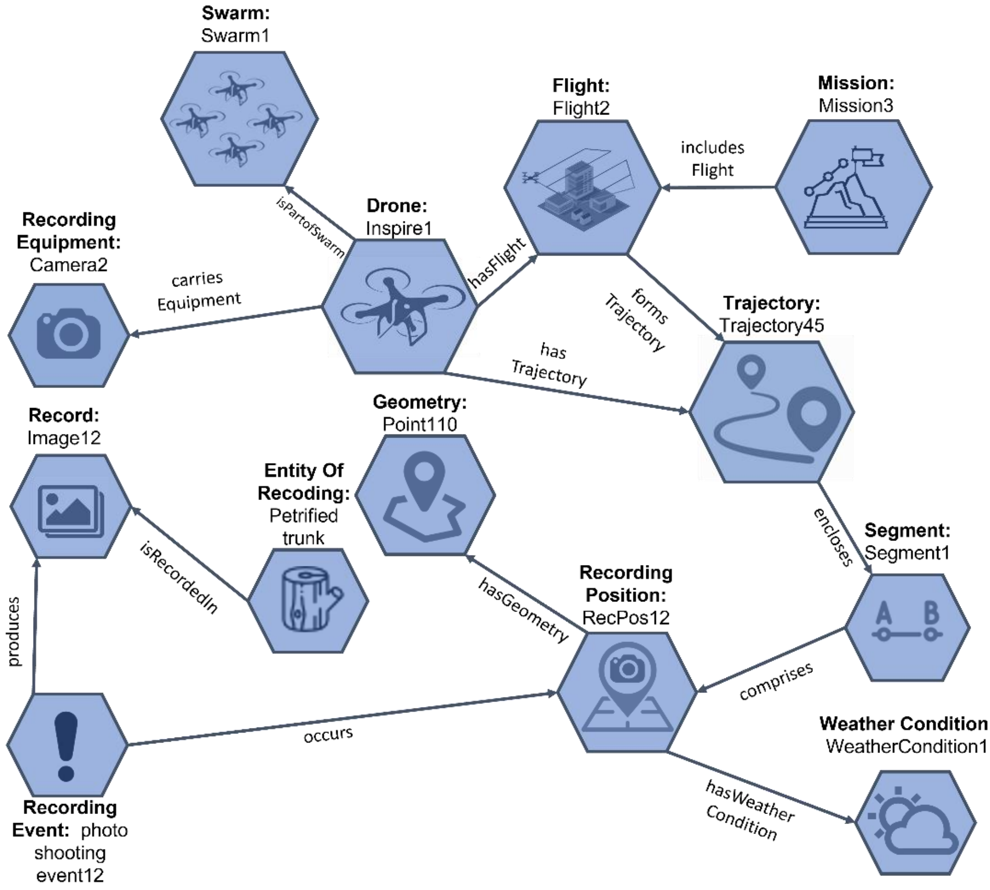


Figure 2. Graph representation of a UAV documentation flight over the Geopark of Lesvos.

A KG is a large semantic network that integrates heterogeneous information sources in order to represent knowledge related to specific domains [6]. It is a graph of data structured to accumulate and convey knowledge of the real world, whose nodes represent entities and edges represent relations between these entities [4]. A KG integrates information into an ontology and applies a reasoning mechanism/engine to infer new knowledge [3]. In cases where a KG builds up quantified statements, ontologies or rules are required in order to provide a more expressive and standard representation of knowledge, while deductive methods can be used to entail further knowledge. Inductive methods can also be used in order to extract additional knowledge from a KG, based on simple or quantified statements [4].

Santipatakis et al. [1] propose the datAcron ontology for representing semantic trajectories at varying levels of spatiotemporal analysis. Mobility analysis tasks are based on a wealth of disparate and heterogeneous sources of information that need to be integrated. The proposed ontology, as a generic conceptual framework, tackles this challenging problem. The experimental results in the air traffic management domain demonstrate that the proposed ontology supports the representation of trajectories at multiple and interlinked levels of analysis. Gao et al. [7] proposed a representation of semantic trajectories that considers domain knowledge, in addition to spatiotemporal data, to achieve improved retrieval of semantic trajectories. The developed method proposes a tree-shaped hierarchical network that detects ROIs in a set of trajectories in order to replace those trajectories with sequences of the detected ROI. The ROI sequences are transformed, based on the geographical and semantic trajectory features, to continuous vectors. The model measures similarities among the vectors and emphasizes the context of trajectories to extract semantic relations among target objects.

The aim of this paper is to present a KG-based integrated UAV approach for engineering cultural heritage semantic trajectories as KGs (STaKG). Specifically, the paper contributes the following:

(a) An updated methodology for the engineering of semantic trajectories as KGs (STaKG);
(b) An implemented toolset for the management of KG-based semantic trajectories;
(c) A refined ontology for the representation of knowledge related to UAVs' semantic trajectories and to cultural heritage documentation;
(d) The application and evaluation of the proposed methodology, the developed toolset, and the ontology within the domain of UAV-based cultural heritage documentation.

The presented work is unique since, to the best of our knowledge, another integrated UAV approach for engineering cultural heritage semantic trajectories as KGs for documentation purposes does not exist. However, related works that individually propose related technologies used for the implementation of the proposed framework/approach are presented in the following sections.

This paper is an extension of our recently published preliminary work on engineering UAVs' semantic trajectories as knowledge graphs [8]. This extension comprises beyond 70% of new content, and it concerns (a) an updated/refined ontology for the representation of the related generic and domain knowledge, (b) the implementation of the proposed STaKG toolset, and (c) the evaluation of the proposed STaKG methodology and toolset with real datasets.

The structure of the paper is as follows: Section 2 presents the materials and methods of the approach, Section 3 presents the results of the evaluated implementation of the approach and a critical discussion on the findings, and Section 4 concludes the paper with a discussion on obtained results and future research plans.

2. Materials and Methods

2.1. STaKG Methodology

Ontologies provide the formal and explicit semantics that KGs need for the effective modeling of linked data, and are considered the backbone of KGs. Ontology engineering methodologies (OEMs) define specific methodological phases, processes, and tasks for the engineering of ontologies, including feasibility analysis, identification of goals, requirements specification, implementation, evaluation, and maintenance. Such engineering methodologies include—to some extent—analogous building and managing tasks for the engineering of KGs. As already presented in a related work [5], an ontology and a KG can both be developed following the general principles and analogous tasks of an OEM, especially those that are based on the iterative collaboration of multiple actors, such as the OEM of DILIGENT [9] and HCOME [10]. Based on this assumption, our work on a methodological approach for engineering KGs borrows and adapts principles and tasks of a collaborative and iterative OEM, engaging actors with different backgrounds and expertise. Such actors form a KG engineering team that assigns three main roles to each member, i.e., the role of the knowledge engineer, the role of the knowledge worker, and the role of the domain expert. For more details, please refer to the reference articles for the DILIGENT [9] and HCOME [10] methodologies.

In this line of research, we present a novel methodology for engineering semantic trajectories as KGs, namely the semantic trajectories as knowledge graphs (STaKG) methodology, inspired and mainly influenced by the Human-Centered Collaborative Ontology

Engineering Methodology (HCOME) [10]. To better reflect the specific engineering requirements of KGs, we selectively borrow principles from other related KG engineering approaches [6,11], fusing them with those of HCOME. The STaKG methodology follows a human-center/top-down and data-driven/bottom-up (hybrid), collaborative, and iterative approach for engineering KGs, supporting the already well-established for ontologies' lifecycle, i.e., specification, development, and evaluation/exploitation.

As depicted in Figure 3, the STaKG methodology introduces three phases, followed in an iterative manner, each having a number of specific tasks assigned to each member of the collaborative engineering team (based on its role in the engineering process).

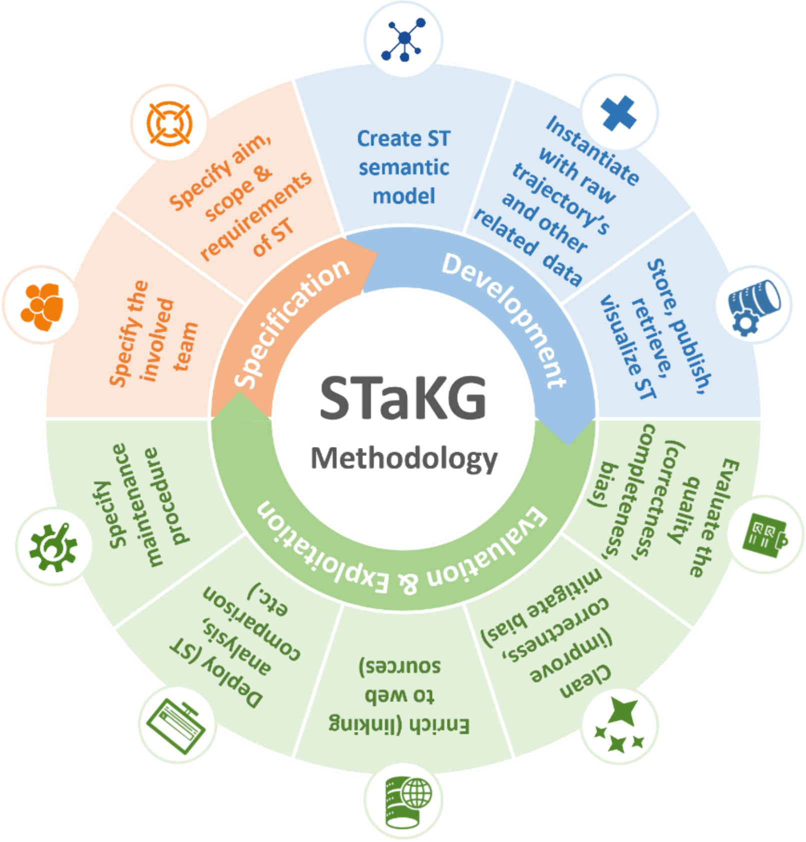

STaKG
Methodology
Specification
Development
Evaluation & Exploitation
Specify the involved team
Specify aim, scope & requirements of ST
Create ST semantic model
Instantiate with raw trajectory's and other related data
Store, publish, retrieve, visualize ST
Evaluate the quality (correctness, completeness, bias)
Clean (improve, correctness, mitigate bias)
Enrich (linking to web sources)
Deploy (ST analysis, comparison, etc.)
Specify maintenance procedure

Figure 3. The STaKG methodology (ST refers to a sematic trajectory).

The specification phase (Figure 3, orange arc) includes the specification of the involved stakeholders and roles in the engineering team (who is doing what), as well as the aim, scope, and requirements in terms of data, semantic annotations, segmentation of the trajectories, and the ontological model that will capture the required knowledge.

The development phase (Figure 3, blue arc) includes the creation of the explicit knowledge related to the STaKGs, i.e., the extension and specialization of reused semantic trajectory models (e.g., an extension of existing semantic trajectory ontology) based on the requirements of the first phase. It also includes the creation of instance data, i.e., spatiotemporal and contextual data related to the recorded trajectories. In the same phase, storage, publishing, retrieval, and visualization tasks related to the STaKGs are performed.

The evaluation/exploitation phase (Figure 3, green arc) includes (a) the evaluation of the quality of the modeled STaKGs in terms of correctness, completeness, and fairness/bias, and (b) the cleaning and enrichment of the STaKGs. Enrichment refers to the discovery of, and linking to, additional/external knowledge sources (e.g., from the web). In the same phase, deployment and maintenance tasks are included.

The exploitation of STaKGs involves a set of services based on the proposed methodology. These services may be useful in several different use cases. A key set of exploitation services include the following:

- Visualization service: provides visualization of trajectories on geographic maps and timelines of events or phenomena, e.g., providing a geographic map that visualizes, in the form of semantic trajectories shaped by data-recording episodes, different flights of a UAV during a specific cultural heritage documentation mission on a specific flight zone and time interval.
- Querying service: provides a multimodal (template-driven and formal language) querying interface for the retrieval of semantic trajectories based on several criteria-variables that are mapped to the conceptual model/ontology used for their representation as KGs, e.g., retrieving the movement, weather, and recording (e.g., photos) data of all UAV documentation flights on a cultural heritage mission that match a specific geographic area (flight zone) and time period (time interval).
- Analytics service: provides analysis of trajectories based on spatiotemporal semantics, e.g., analyzing the duration and recordings of a specific flight of a UAV in a specific area of interest. Analytics are realized through the following tasks:
    - ○ Comparing semantic trajectories in terms of spatiotemporal criteria, e.g., the semantic trajectories of two recording episodes for the same recording points (POIs) or space (flight zone);
    - ○ Merging semantic trajectories, e.g., merging semantic trajectories that occur in the same recording mission of a specific UAV;
    - ○ Split semantic trajectories to specific episodes, e.g., splitting the trajectory of a UAV flight into episodes related to the camera-shooting position (set at up–shooting–departure for the next shooting position);
    - ○ Discovering the behaviors of moving entities (behavior analytics) where there is no previous knowledge for the behavior, e.g., discovering types of flights of UAVs based on the flight behaviors/patterns followed by their operators, or recognizing the aim of the flight or the mission (surveillance flight/scientific flight) based on the semantic trajectory of a drone and the carried equipment;
    - ○ Evaluating semantic trajectories in terms of spatiotemporal information and its correlation with other contextual data, e.g., evaluating the expected efficiency (e.g., duration, altitude) of a flight or mission based on the environmental conditions of the flight or mission (e.g., high efficiency cannot be expected during bad weather conditions).

2.2. STaKG Knowledge Model

The presented work was originally designed to answer a number of competency questions shaped during the specification phase of STaKG engineering methodology, with the participation of domain experts (geography, geoinformatics, and UAV pilot experts). Specifically, we interviewed experts in the field of UAV-based documentation, along with experts in documenting cultural heritage sites, in order to obtain the information/knowledge required for engineering the ontological model and for designing the toolset. Based on these interviews, a number of competency questions were shaped collaboratively with all stakeholders. A representative list of the questions that were used to evaluate the developed toolset is provided below:

- Which trajectories of a specific mission include records of a specific object?
- Which recording positions include records of a specific object?
- What kind of records are produced during a specific mission?
- Which missions result in photograph records?
- What are the recording positions of a specific flight?
- What kind of records are produced at a specific recording position?
- What are the recording segments of a trajectory?
- What are the weather conditions at a specific point in time for a specific flight?

- Which flights intersect?
- What is the number of drones involved in a specific mission and the number of flights initiated for that mission?
- What recording events occurred at a distance of less than 100 m from a specific recording event?
- Which recording events took place near a specific POI?

The data and knowledge that the underlying model of an STaKG aims to capture is derived from five different—though interlinked—thematic domains: (i) trajectory information, (ii) drones' flight and mission information, (iii) recording events and resulting records, (iv) geographical information of regions/points of interest, and (v) weather conditions. The data types related to the five thematic domains are the following:

- Flight data, derived from flight log files, which are written records of a flight automatically generated by a drone. Flight log files contain flight details concerning flight planning information along with time-stamped movement of the drone and on-board sensor data (e.g., longitude, latitude, altitude, timestamp of different positions). Flight log files are usually stored (usually in CSV format) in the native application of a device (remote control, mobile phone, or tablet) and the drone's pilot application.
- Equipment data, which are the data reported by the flight operator, describing the characteristics of a drone (e.g., model, serial number, software type). These data are documented after the in situ survey using drone data management software (the drone logbook).
- Mission data, which are the data reported by the flight operator in the context of the mission planning procedure (e.g., the purpose of the mission, the category of the mission, the area of the mission, the equipment to be used). These data are documented by experts right after the mission, using drone data management software.
- Recorded data (aerial and terrestrial), which are data extracted from files (photos, videos, lidar data) acquired during the flight mission (e.g., longitude, latitude, date, time of the recording). These data are provided either by Exif files of the records or directly from the records.
- Geographic names and elements, which are data about the POIs/ROIs at which the objects of interest are located, and where the drone's mission occurs (e.g., cities, villages, ports, buildings, archaeological sites).
- Weather data, which are data (e.g., temperature, humidity, wind velocity) recorded by weather monitoring devices or systems. These they are dynamically collected from external (web) services or/and in-drone sensors, based on the time and location of the mission that is recorded.

For the development of the semantic model that is required to represent knowledge related to STaKGs, existing related models were studied, such as the datAcron [12], Dronetology [13], and the W3C Geospatial Ontologies [14]. These models were selected based on our previous experience in related work, as well as a search of ontology repositories (LOV [15] and ODP [16]) using terms such as "trajectory", "drone", "weather", "recording", and "record". At this point, the datAcron ontology was selected to reuse the main conceptualization of the semantic trajectory and the aviation, which are both included in the scope of the semantic model under development. The datAcron ontology imports the ontologies (i) DUL (DOLCE + DnS Ultralite ontology) [17], which is a simplification of the DOLCE Lite-Plus library, (ii) SKOS [18] (Simple Knowledge Organization System), which is a data model for sharing and linking knowledge organization systems (e.g., classification schemes, authority files, subject headings) via the web, (iii) the SOSA/SSN [19] (Sensor, Observation, Sampler, and Actuator/Semantic Sensor Network), which describes the context of sensor and actuator activity (including systems of sensors and actuators, observations, procedures, subjects, samples, etc.), (iv) SF (Simple Features Geometry) [20], which defines Simple Feature geometry types (a set of standards for the specification of geographic features used by geographic information systems [21]), (v) GML [22] (Geography Markup Language), which is an XML encoding for the transport and storage of geographic information, and

(vi) GeoSparql [23], which is an ontology for spatial information and includes SPARQL extension functions and RIF rules supporting queries and reasoning.

In the context of our case study, it was necessary to represent the thematic category that a subject of recording presents. For example, it needs to express that an entity of interest—which has been recorded—has archaeological interest and therefore is included in the category "Archaeology". To accomplish a consistent thematic annotation that will be meaningful for the domain of interest, as well as to reuse existing terminology, which is available in an SW technology format, we chose a version of the UNESCO SKOS Thesaurus [24] proposed by the Greek National Documentation Center. It is an SKOS Thesaurus that focuses on terms within the cultural, educational, and social and humanities domain, provided in the Greek language and continuously updated by Semantics.gr [25]. Semantics.gr is the national semantic content aggregator and search platform developed and hosted by the Greek National Documentation Center, which supports the publishing of documented resources as linked open data. The thesaurus was directly imported in the developed ontology.

The working version of our semantic model (Onto4drone [26]) is available in OWL and accessible online at https://github.com/KotisK/onto4drone (accessed on 12 January 2023). The model was developed following the HCOME collaborative engineering methodology, supported by Protege 5.5 (for personal space engineering), and WebProtégé (for shared space engineering), respectively. In addition, Google Docs and Google Meet were used for further collaborative engineering tasks.

The basic concepts and relations of the model implemented in the Onto4drone ontology are briefly presented in Figure 4 in the form of a concept map.

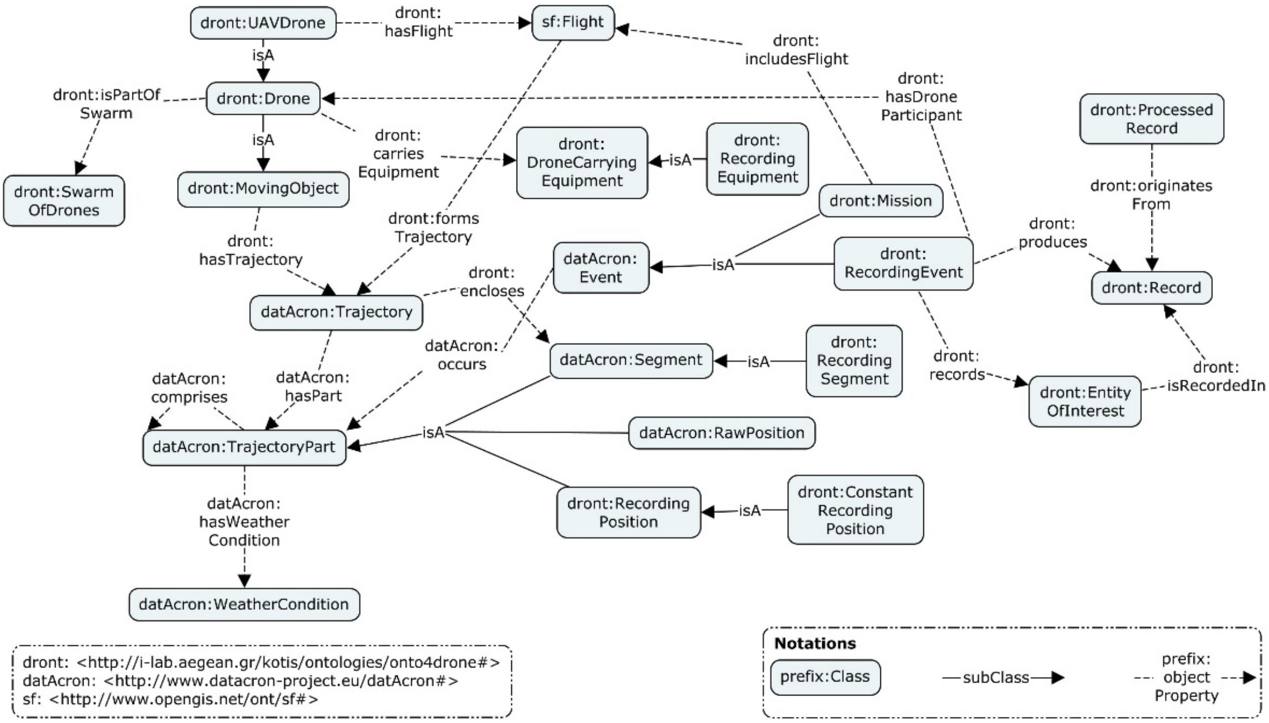


Figure 4. Basic concepts and relations of the STaKG ontological model representing drones and recording episodes.

It is worth mentioning that some of the concepts, such as the TrajectoryPart concept, may be perceived as "general" because they are not specialized for the particular case study (UAV/drone recording). Those concepts are mainly imported and reused by the existing ontologies (namely datAcron, GeoSparql) that were identified in the first steps of the semantic model development. Particularly, the classes Time, MovingObject, Trajectory, TrajectoryPart, WeatherCondition, Geometry, Flight, Box, Event, Segment, TrajectoryCluster, SpatialObject, and Point have been reused.

On the other hand, more specialized concepts, which were identified in the first steps of the development of the semantic model, have been added, and in some cases extend the more general concepts in order to satisfy the requirements of the case study. For instance, specialized classes such as RecordingPosition—a subclass of the TrajectoryPart class—are introduced in the semantic model in order to define parts of the trajectory where recording (instantly or for a particular duration) occurs. Table 1 presents a set of key classes of the developed ontology.

Table 1. Key classes of the Onto4drone ontology.

Super Class Reused from Other Ontologies	Class in Onto4Drone
dul:Entity	EntityOfInterest
dul:Abstract	Swarm
datAcron:TrajectoryPart	RecordingPosition
datAcron:Segment	RecordingSegment
dul:Entity	EventOfInterest
datAcron:Event	MilitaryEvent
datAcron:Event	Mission
datAcron:Event	RecordingEvent
dul:Object	ObjectOfInterest
datAcron:Document	ProcessedRecrod
datAcron:Document	Record
datAcron:MovingObject	Drone
datAcron:MovingObject	FlyingObject
datAcron:MovingObject	MarineObject
datAcron:Point	PointOfInterest
datAcron:Region	RegionOfInterest

Furthermore, there are relations (object properties) that have been reused in our model, such as hasPart, comprises, occurs, hasGeometry, hasTemporalFeature, hasWeatherCondition, hasLowPoint, hasHighPoint, overlaps, contains, touches, hasIntervalDate, timeEnd, timeStart, hasLatitude, hasLongitude, and hasVelocity. Finally, there are newly introduced relations based on the requirements of the case study, such as the relation resultsIn, which correlates a flight or mission with the produced records, and the relation occurs, which correlates a recording event with the episode or position where the recording took place. Some of the introduced object properties are sub-properties of reused ones. These properties, along with their domain and range classes, are presented in Table 2.

The current version of Onto4drone includes classes, object properties, and data properties based on the motivated scenarios of documenting cultural heritage sites using UAV recordings. For example, a representative restriction is one that indicates a RecordingEvent has occurred in at least one RecordingPosition or RecordingSegment (Equation (1)).

$$\text{RecordingEvent} \subseteq \exists \text{occurs.}(\text{RecordingPosition} \cup \text{RecordingSegment}) \tag{1}$$

Furthermore, a small set of rules were developed in order to evaluate inferencing for the segmentation of trajectories. We used SWRL rules [27], as formulated within the Protégé plugin SWRLTab (https://protegewiki.stanford.edu/wiki/SWRLTab). One example of an SWRL rule is the following, which infers that if a recording event has occurred in a position within the trajectory of a drone, then this is a recording position (Equation (2)):

dront:RecordingEvent(?a) ˆ datAcron:occurs(?a, ?b) ˆ datAcron:RawPosition(?b)
-> dront:RecordingPosition(?b) (2)

Table 2. Key properties of the Onto4drone ontology.

Object Property	Domain	Range	Super Object Property
dront:includesFlight	dront:Mission	sf:Flight	dul:hasConstituent
dront:clusters	dront:TrajectoryCluster	datAcron:Trajectory	dul:hasPart
dront:hasPartDrone	dront:SwarmOfDrones	dront:Drone	dul:hasPart
dront:encloses	datAcron:Trajectory	datAcron:Segment	datAcron:hasPart
dront:hasDroneParticipant	dront:RecordingEvent or dront:MilitaryEvent	dront:Drone	dul:hasParticipant
dront:isPartOfSwarm	dront:Drone	dront:SwarmOfDrones	dul:isPartOf
dront:hasFlight	dront:UAVDrone	sf:Flight	dul:isParticipantIn
dront:isDroneParticipantIn	dront:Drone	dront:RecordingEvent or dront:MilitaryEvent	dul:isParticipantIn
dront:originatesFrom	dront:ProcessedRecord	dront:Record	dul:associatedWith

In order to evaluate the engineered model at this initial stage, a number of individuals were established to populate the ontology. Additionally, a set of SHACL rules [28] were formulated in order to evaluate the individuals. Regarding SHACL validation and constraint rule formulation, the Protégé plugin SHACL4Protege (https://github.com/fekaputra/shacl-plugin) was used. For instance, the following rule defines the expected relations for a recording event instance:

```
dront:RecordingEventShape
    a sh:NodeShape ;
    sh:targetClass dront:RecordingEvent ; # Applies to all recording events
    sh:property [             # 1
    sh:path datAcron:occurs ; # constrains the values of datAcron:occurs
        sh:class dront:RecordingPosition ;
        sh:maxCount 1 ;
    ] ;
    sh:property [             # 2
        sh:path dront:records ;             # constrains the values of dront:records
/small  sh:class dront:EntityOfInterest ;
        sh:minCount 1 ;
    ] ;
    sh:property [            # 3
        sh:path dront:produces ;       # constrains the values of dront:produces
        sh:class dront:Record ;
        sh:minCount 1 ;
    ] ;
    sh:property [                 # 4
        sh:path dront:hasDroneParticipant ;          # constrains the values of
dront:hasDroneParticipant
```

```
            sh:class dront:Drone ;
                    sh:maxCount 1 ;
                ] ;
                sh:closed true ;
/small
                sh:ignoredProperties (rdf:type owl:topDataProperty owl:topObjectProperty
            rdfs:label dul:hasParticipant) ;
```

Finally, in the context of the semantic model development, a number of competency questions (CQs) were formulated, in collaboration with the domain experts, which are questions that must be answered by querying the semantic model and the KG. CQs are useful both for the semantic model and KG building, as well as for their evaluation. The CQs are aligned to graph-based queries, e.g., SPARQL [29] or Cypher queries, which were used for evaluation (see Section 4, Discussion). For instance, using the ontology, it is possible to answer the question (via querying the available data) "which trajectories include recording events that have recorded Petrified Trunks?":

```
SELECT ?trajectory
      WHERE { ?recordingEvent dront:records dront:petrifiedTrunk.
            ?recordingEvent datacron:occurs ?recordingPosition.
            ?trajectorySegment datacron:comprises ?recordingPosition.
            ?trajectory dront:encloses ?trajectorySegment}.
```

2.3. STaKG Toolset

To support the STaKG methodology with an engineering environment, we designed and developed a toolset based on state-of-the-art technology for linked data and KGs. Its interconnected components exchange data through a pipeline process (Figure 5) that involves the following aspects:

(a) The preprocessing of position/movement data (data cleaning, data compression);
(b) The conversion of raw trajectories to STs via applying semantic annotation based on the semantic trajectory model (onto4drone ontology);
(c) STaKG management and retrieval;
(d) Enrichment of STaKGs (connection to related KGs, utilization of application domain and geographical data);
(e) Analysis of STaKGs to recognize semantic behaviors (classification, clustering, aggregation, comparison of STaKGs).

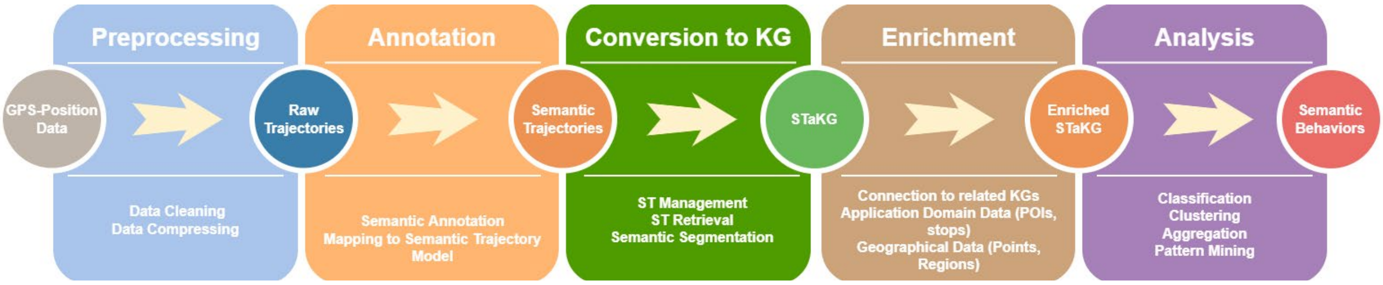


Figure 5. The STaKG pipeline (framework).

The high-level architecture of the designed toolset (Figure 6), supporting the pipeline introduced in Figure 5, includes the following elements:

(a) A tool for raw trajectory data cleaning and RDFization based on automated/semi-automated mapping to related semantic models, utilizing specific tools (OpenRefine [30], Karma [31], Neo4j [32]);
(b) A tool for trajectory data summarization and enrichment with recording metadata, weather data, and structured data of POI/ROI shape files;
(c) A tool for semantic trajectory management (split, merge, combine, analyze);
(d) A web-based tool for semantic trajectory browsing and visualization.

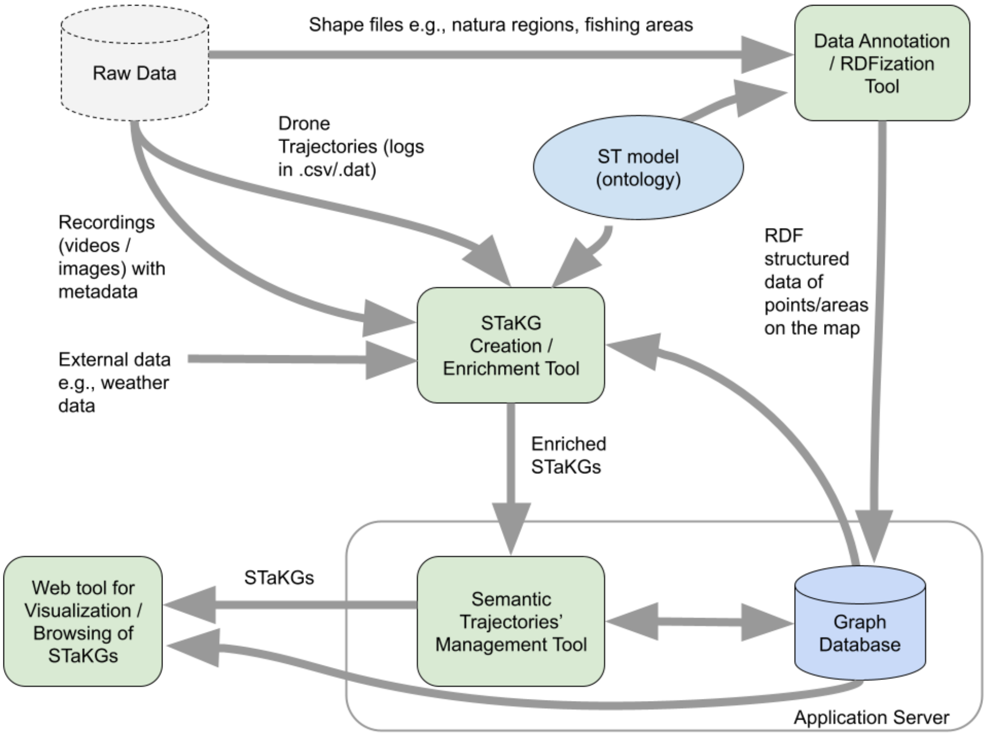

Raw Data
Shape files e.g., natura regions, fishing areas
Data Annotation / RDFization Tool
Drone Trajectories (logs in .csv/.dat)
ST model (ontology)
Recordings (videos / images) with metadata
RDF structured data of points/areas on the map
STaKG Creation / Enrichment Tool
External data e.g., weather data
Enriched STaKGs
STaKGs
Web tool for Visualization / Browsing of STaKGs
Semantic Trajectories' Management Tool
Graph Database
Application Server

Figure 6. High-level architecture and data exchange of STaKG toolset.

The tools described in (b), (c), and (d) were developed using Neo4j.rb (http://neo4jrb.io/), React (https://reactjs.org/), Neodash (https://neo4j.com/labs/neodash/), Neo4j (https://neo4j.com/), and Neosemantics (https://neo4j.com/labs/neosemantics/4.0/). The latter (Neosemantics) is a tool designed to utilize integrations between RDF and LPG graphs [32]. In the developed application, it loads the Onto4drone ontology to the KG, provides functionality for loading RDF data to the KG in case of further enrichment with linked open data, and exports the data in the RDF format. Furthermore, a graph database, namely Neo4j [33], supports the deployment of the web-based tool, and stores the managed data (semantic trajectories, GIS recording missions, and produced records). Especially for RDF store technology, although noteworthy alternatives exist that are specialized in spatiotemporal RDF data storing, e.g., Strabon RDF store [34], Neo4j was selected due to the integrated graph analytics solutions that it provides.

A high-level architectural design of the interconnected tools of the STaKG toolset and the related exchanged data are depicted in Figure 5. Raw archive data including drone flights' log files, metadata of recordings, and shapefiles of geographical regions were annotated and used for the creation and enrichment of the STs, along with other external open data, such as weather data (Historica Weather API https://open-meteo.com/en/docs/historical-weather-api) and POIs (Open Street Maps Overpass API http://overpass-api.de/api/interpreter). The annotation and enrichment processes are based on the ST model. Enriched STs are stored in the graph database as a KG. The ST management tool uses STaKGs to create trajectory segmentations and perform analytics and tasks such as merging, splitting, and combining trajectories. The web tool for visualization and ST browsing fetches analytics results and STaKG data stored in the graph database through predefined and customizable queries, in order to efficiently present them to the user.

The latest version of the developed toolset is accessible online at https://github.com/sotirisAng/stakg.

3. Results

3.1. Use Cases and Correspondent Datasets

Three-dimensional mapping of geographical areas that attract the interest of both scientists and the public requires the utilization of various UAVs with different recording sensors. In the present research, data were acquired from two UAVs: (a) Phantom 4 Pro and (b) Inspire 2. The Phantom 4 Pro is a mid-range quadcopter that features a 20 mpxl camera with a focal length of 9 mm and a sensor size of 13.2 × 8.8 mm. The Inspire 2 is a mid-range quadcopter with a total weight of 3.290 kg. The recording sensor used was a Zenmuse X5S camera with a 25 mm focal length Olympus lens. The data acquired consist of very high resolution images and 4K videos accompanied by metadata that describe the technical characteristics and information of the UAVs and sensors during the flight. Data acquisition was performed via flight plans and manual flights. Samples of the datasets (in CSV and RDF syntax) are accessible online at https://github.com/sotirisAng/stakg.

3.1.1. Petrified Forest

The Lesvos Petrified Forest (Geopark) includes many individual fossil sites, as well as organized visitor parks. The Lesvos Petrified Forest parks are areas of high geological importance, and 3D mapping these areas at regular intervals is critical as it forms the basis for conservation, protection, and discovery of fossils. High-resolution images were acquired with the Phantom 4 Pro. The first flight of the Phantom 4 Pro at the Lesvos Petrified Forest was designed in the Mission Hub Litchi software. The UAV flew at an altitude of 25 m above the ground for the purpose of 3D mapping fossil sites situated on the first flight in Balli Alonia Park. The second flight of the Phantom 4 Pro was performed at a fossil site along the new Kalloni–Sigri major road. The flight was conducted manually as the extent and depth of the site was altered due to excavation processes at the site. The visualization of the trajectory formed by the flight is depicted in Figure 7. The following table (Table 3) presents the details of the two flights.

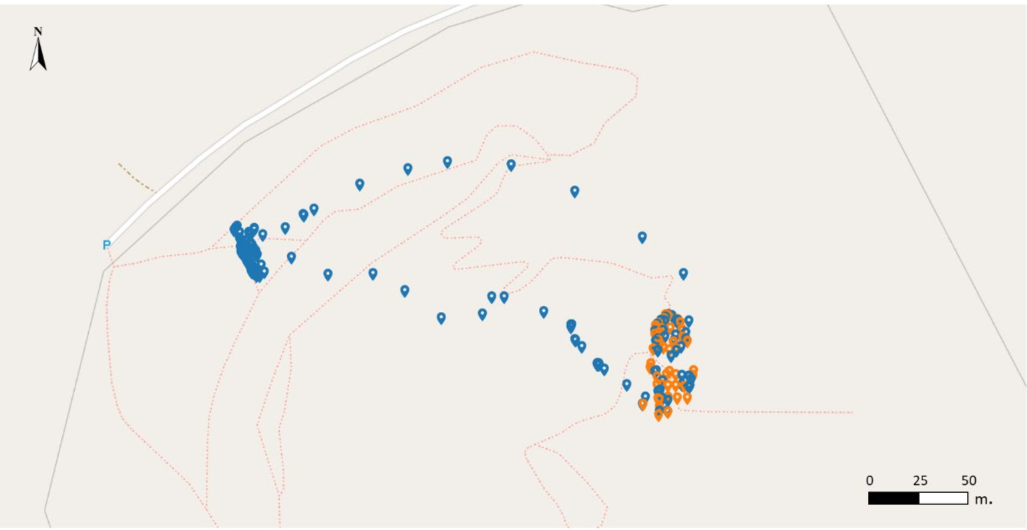


Figure 7. Visualization of trajectory of flight over the Sigri Geopark. Blue-colored points represent raw positions and orange-colored points represent recording positions.

Table 3. Flight details for Lesvos Petrified Forest dataset.

UAV	Flight Altitude	Overlap	Flight Duration	Software	Log File Data Format
Phantom 4 Pro	25 m	90% and 60%	13 min	Litchi mission hub	.csv
Phantom 4 Pro	5–12 m	80% and 70%	7 min	Litchi mission hub	.csv

3.1.2. Vrissa Village

The settlement of Vrissa is located on the island of Lesvos, in the northeastern Aegean Sea in Greece. It has been characterized as a traditional settlement by the Greek government and is protected by relevant decrees. An earthquake significantly affected the settlement of Vrissa. Many homesteads collapsed and others sustained heavy damage. Due to this emergency, a high-resolution 3D recording of the existing state of the settlement was deemed necessary a few days after the devastating earthquake. In addition, spatiotemporal monitoring of the settlement was implemented to record any future changes. In more detail, for the purposes of this study, data were acquired with the Phantom 4 Pro. Both flight plans were designed in the Litchi Mission Hub software with a flight altitude of 80 m (Table 4). The flights aimed at the overall 3D mapping of the settlement of Vrissa two years after the devastating earthquake. The flight pattern was designed in a grid format, and the images overlapped 80% at the front and 80% at the side with the aircraft camera pointed vertically at the ground (−90°). The second flight, performed three years after the earthquake, also aimed at the spatiotemporal monitoring of the settlement. The front overlap was 90% and the side overlap was 60%. The UAV camera was set vertical to the ground to produce a high-resolution orthomosaic of the area. The visualization of the trajectory formed by the flight is depicted in Figure 8 The flights' metadata were exported in .csv format from the Mission Hub Litchi application.

Table 4. Flight details for Vrissa dataset.

UAV	Flight Altitude	Overlap	Flight Duration	Software	Log File Data Format
Phantom 4 Pro	80 m	80% and 80%	21 min	Litchi mission hub	.csv
Phantom 4 Pro	80 m	90% and 60%	16 min	Litchi mission hub	.csv

3.1.3. University Hill

Two flights at the building of the Department of Geography of the University of the Aegean in Lesvos were performed for the research purposes outlined in this article, with the Phantom 4 Pro and the Inspire 2 UAVs. The flight details are presented in Table 5. For the first flight, the Inspire 2 captured the perimeter and the buildings that form the campus. The flight height was varied from 5 m to 50 m from the take-off point. The second flight was performed manually with the Phantom 4 Pro. The aircraft flew at a height of 7 to 60 m from the take-off point. The visualization of the trajectory formed by the flight is depicted in Figure 9. The launch site was the roof of a building for both UAVs. The areas of interest mapped by both UAVs were the buildings of the University of the Aegean located on University Hill. The flights were operated through the DJI application, GO 4, whose log files are in .dat format.

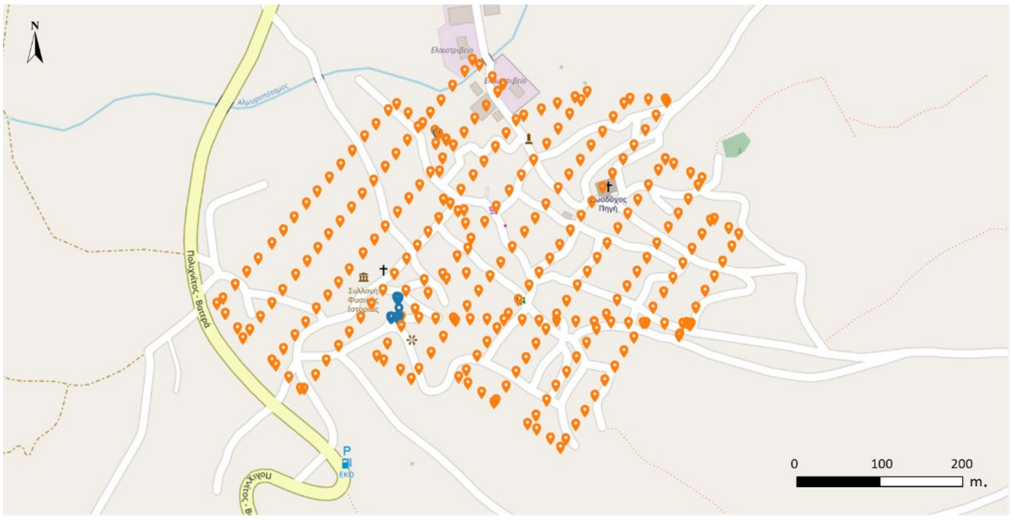


Figure 8. Visualization of trajectory of flight over the village of Vrissa. Blue-colored points represent raw positions and orange-colored points represent recording positions.

Table 5. Flight details for University Hill dataset.

UAV	Flight Altitude	Overlap	Flight Duration	Software	Log File Data Format
Phantom 4 Pro	7–60 m	Manual flight	3 min	Dji Go 4	.dat
Inspire 2	5–50 m	Manual flight	4 min	Dji Go 4	.dat

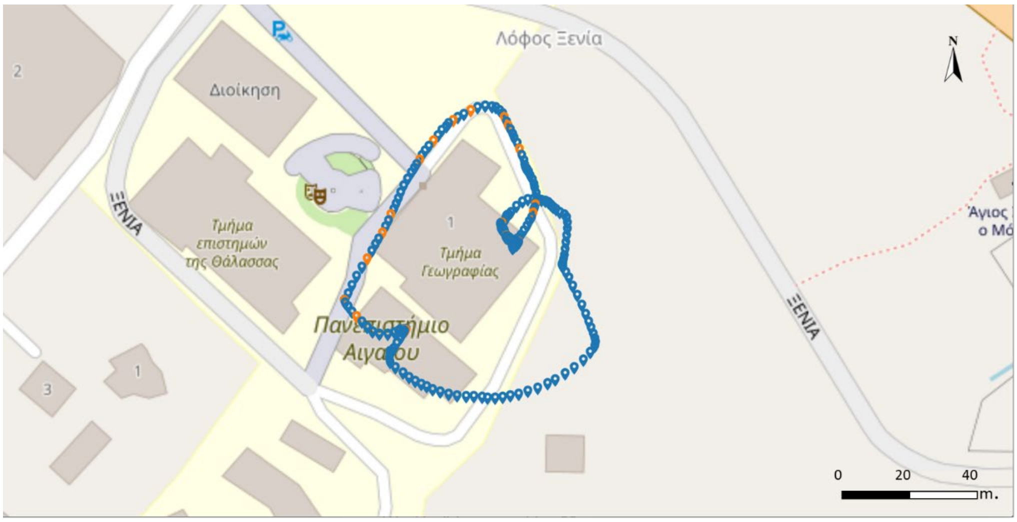


Figure 9. Visualization of trajectory of flight over the University of the Aegean. Blue-colored points represent raw positions and orange-colored points represent recording positions.

3.1.4. Geographical Dataset

The University of the Aegean (UoA) produces and maintains a significant amount of high-quality and high-resolution geographical datasets for the Aegean Sea and the wider area. These datasets cover a variety of thematic categories, including administrative boundaries (borders, municipalities, etc.), physical characteristics (caves, volcanoes, petrified trunks., etc.), natural hazards (floods, earthquakes, etc.), culture (museums, archeological sites, etc.), infrastructure (airports, road networks, etc.), and services (schools, tax offices, etc.). In order to upgrade its services, the University of the Aegean has taken actions to semantically annotate the above datasets and make them available to the public through an SPARQL endpoint, accessible at http://semantics.aegean.gr (accessed on 12 January 2023) (from SPARQL endpoint link).

3.2. Semantic Trajectory Processing

3.2.1. ST Creation

The first step in the creation of the STs is to load Onto4Drone ontology to the Graph Database using the Neosemantics plugin. The second step is the preprocessing of the CSV log files from drone flights, by running a set of semi-automated scripts in order to:

- Keep only the data required for the use case;
- Map CSV columns of the logs produced from different pilot software (DJI, Litchi) in a unified template;
- Reduce the point density of the GPS data.

The STs were created by executing a Cypher script (Appendix A.1) to parse the preprocessed CSV files and map the data of each record to labeled nodes and their attributes based on the loaded ontology.

3.2.2. ST Enrichment

Weather data

Trajectories are enriched with weather data for the specific area and time range of each drone flight. The weather data are obtained from the Historical Weather API (https://open-meteo.com/en/docs/historical-weather-api). The weather data provided by the API are used to create WeatherCondition nodes, which are connected to trajectory positions.

Records

Metadata that contains geolocation, timestamp, and file name are extracted from records that are produced during the drone flights. These metadata are used to define the recording events that produce the records. The Cypher script (Appendix A.2) enriches the trajectory by creating recording positions and connecting them to records through recording events.

POIs

External sources are utilized in the enrichment process to provide information about POIs that may be recorded in the records produced during the drone flights. For each record, a query is sent to the external APIs to request documented POIs near the location of the drone, when the record was produced. The maximum distance for a close POI is set to 50 m. The API responses contain information about POIs, such as label, type, and unique URI, for example, label “Department of Geography”, type “University”, uri “https://www.openstreetmap.org/node/5389966411”. This information is used to merge POI nodes in the KG and correlate them with record nodes.

Overpass API is used to fetch POIs documented in OpenSteetMaps (OSM). The Overpass query (Appendix A.3) requests the OSM nodes, which are closer than 50 m to a given location, and they have a name attribute.

UoA geographical datasets that are accessible through the UoA SPARQL endpoint (http://semantics.aegean.gr:3030) provide nearby POIs as a result of the execution of GeoSparql queries (Appendix A.4).

Figure 10 depicts a part of the KG that includes a recording event that produces a record. The record is related to a POI that is of the type petrified trunk. The POI node and the class node are both created based on the response from the UoA SPARQL endpoint.

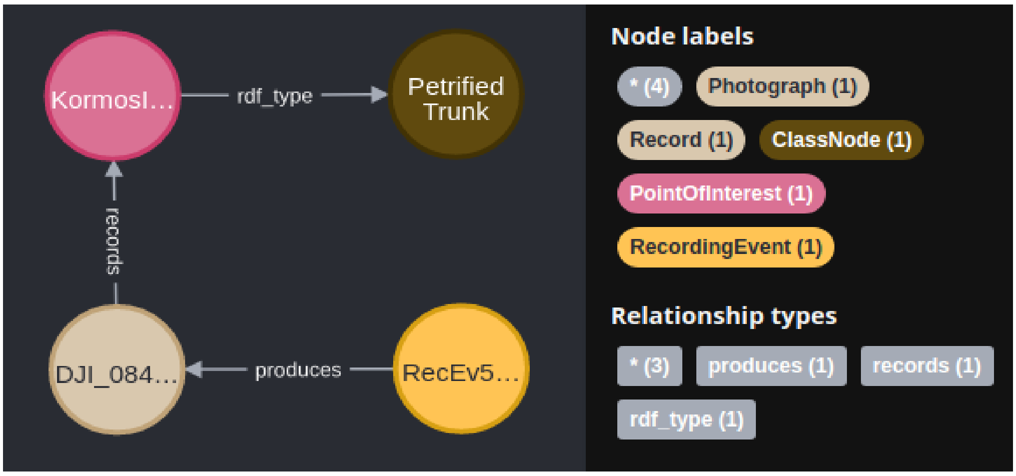


Figure 10. Part of enriched graph with POIs. The * symbol represents the number of nodes (4) and the number of relationships (3) between the nodes.

3.2.3. ST Segmentation

The segmentation that is currently supported by the implemented tool annotates the ST with recording and non-recording segments. This is achieved by inspecting all positions of the trajectory and creating a new recording segment for every position sequence that includes consecutive recording positions that do not have a time difference greater than 15 s. In Figure 11, a segmented ST over the campus of the University of the Aegean in Mytilene is depicted. Recording segments are displayed in orange and non-recording segments in green.

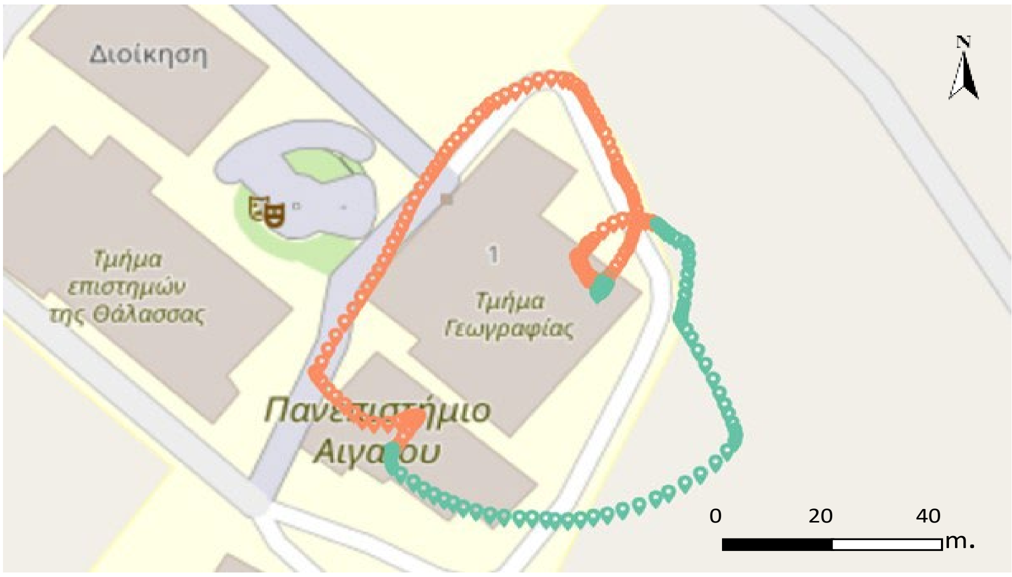


Figure 11. Recording (orange) and non-recording (green) segments of an ST over campus at University Hill.

3.2.4. ST Intersection

In order to discover intersections between trajectories, spatial lines are created from the points included in each trajectory. The spatial function of intersection is applied to each pair of trajectories, and if the function returns intersecting points, a new "intersects" relation is created the trajectory pair.

3.3. ST Visualization

The visualizations tool is based on the Neodash Neo4j plugin, which provides several options for reporting and visualizing graph data, such as Graph, Map, and Table and Chart reports. The reports are populated by the results of cypher queries. Minor amendments have been applied to Neodash in order to enhance the user experience by allowing the passing of information between reports that the users interact with.

The visualization of a trajectory (Figure 12) is achieved by executing a Cypher query (Appendix B.1), which returns the points of all the positions that are part of a specific trajectory.

The visualization of a trajectory and the recording positions (Figure 13) is achieved by executing a Cypher query (Appendix B.2), which returns the points of raw positions (colored blue) and the points of recording positions (colored orange).

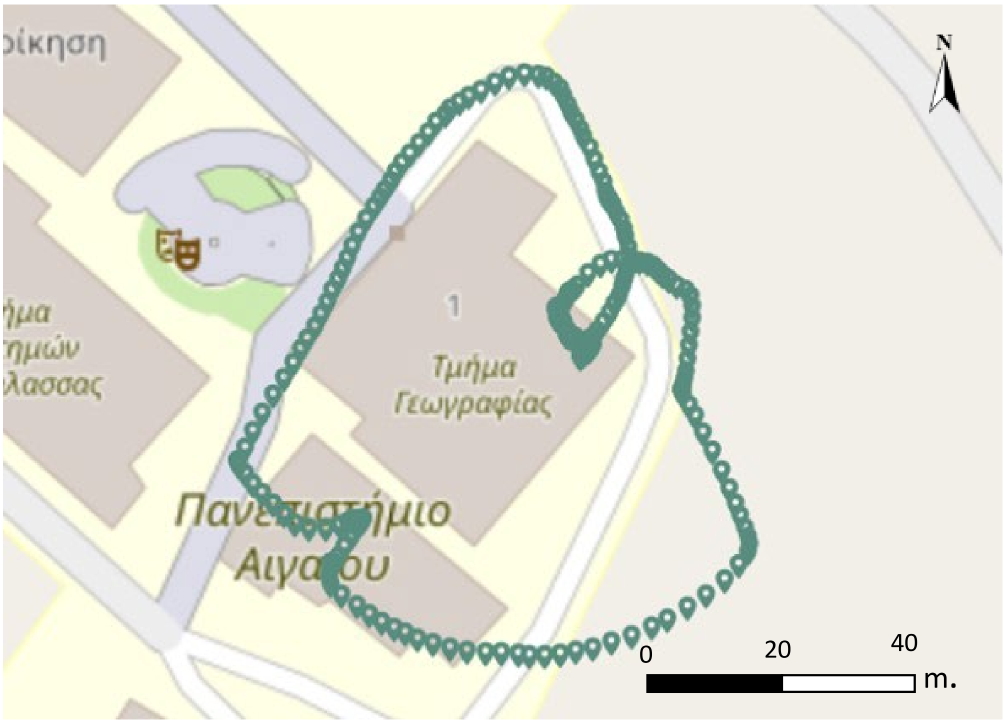


Figure 12. Trajectory visualization (green points).

Each of the recording positions of a trajectory can be selected from the visualization (Figure 13). The selected recording position is passed to a graph visualization, which provides information about the recording events that occur in the selected position. In Figure 14, a graph visualization is provided that displays the results of a Cypher query (Appendix B.3), including the nodes of the recording event, the recording position, the trajectory, and the location point, as well as the record produced by the recording event and the two POIs that the record captures. Hovering over each node enables a tooltip that displays the node attributes. The selection of a record in the graph visualization provides the record source path to the record preview report, which displays the record.

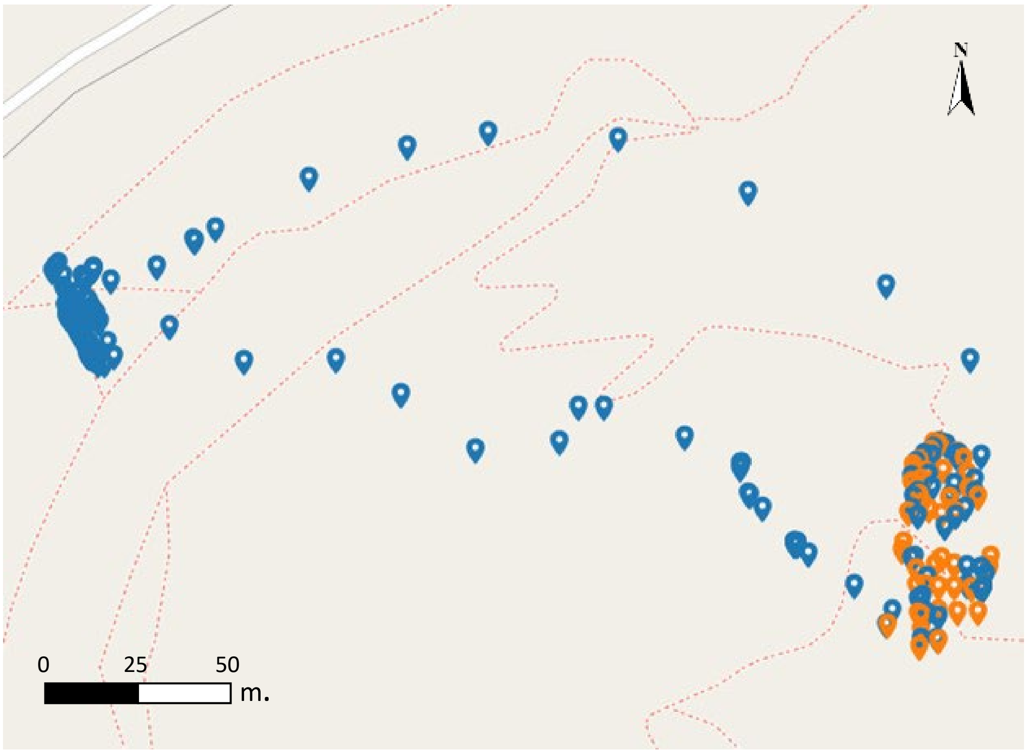
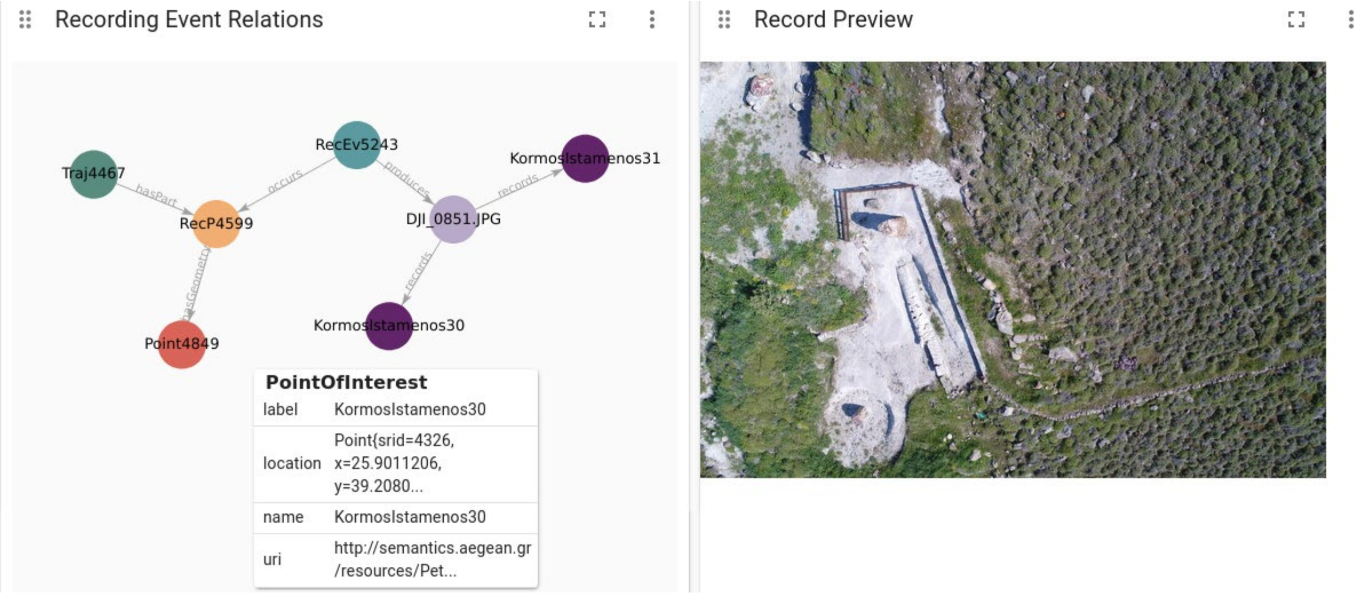


Figure 13. Trajectory visualization of raw (blue) and recording (orange) positions.



PointOfInterest

label	KormosIstamenos30

location	Point{srid=4326,
	x=25.9011206,
	y=39.2080...

name	KormosIstamenos30
uri	http://semantics.aegean.gr /resources/Pet...

Figure 14. Visualization of information regarding a recording event and record preview.

Recording segments of a trajectory (Figure 15) are visualized through a map report that displays the results of a Cypher query (Appendix B.4) for a selected trajectory.

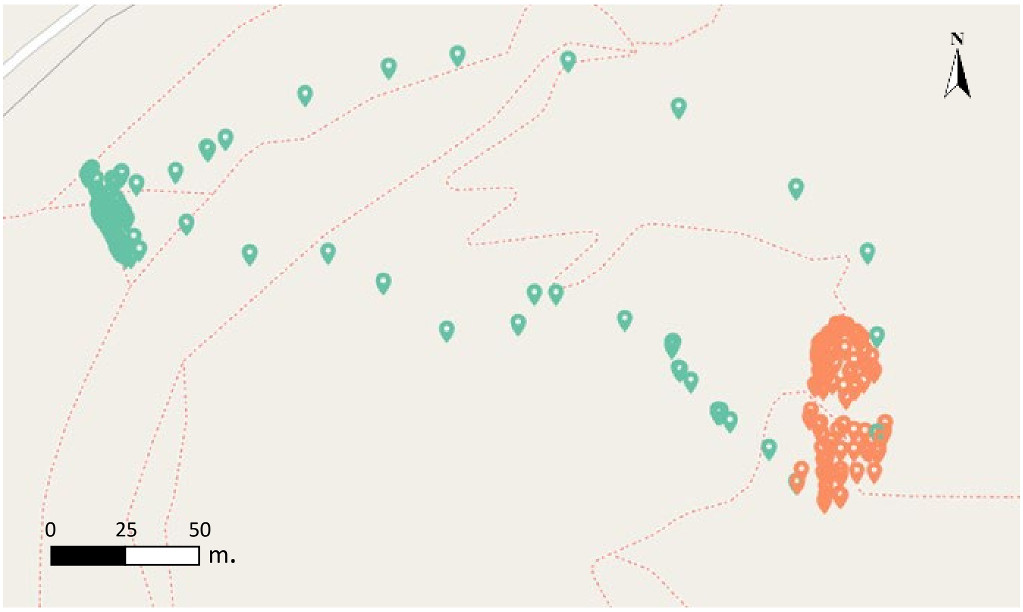


Figure 15. Recording segments of a trajectory (with orange color).

Intersecting trajectories are displayed (Figure 16) on a map report by executing a Cypher query (Appendix B.5).

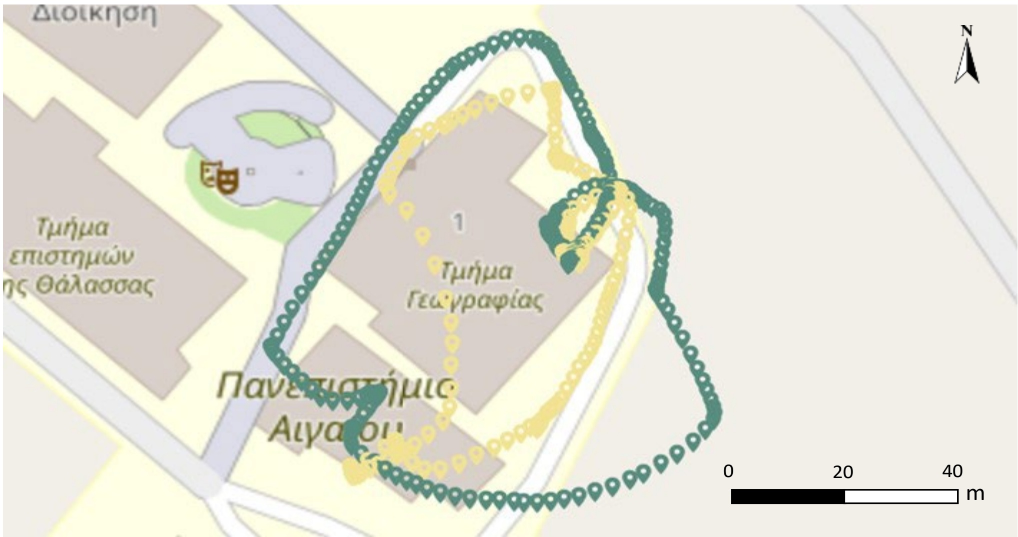


Figure 16. Two trajectories depicted with different colors (green for trajectory A, yellow for trajectory B), intersecting at several points.

3.4. Analytics

A wide range of analytics can be addressed by querying the KG in order to obtain insights regarding the STaKGs. The following list presents a set of representative queries, alongside the Cypher queries and sample results:

- The number of recording points in a trajectory (Appendix B.6).
- POIs in a trajectory (Appendix B.7), e.g., http://semantics.aegean.gr/resources/University_2585.
- Mean temperature recorded during a trajectory (Appendix B.8).
- Number of POIs that are of a specific type in a radius R of a selected point (Appendix B.9).
- POIs that two or more trajectories have in common (Appendix B.10), e.g., http://semantics.aegean.gr/resources/University_2585.

4. Discussion and Future Plans

To the best of our knowledge, the presented work is unique in terms of delivering an integrated UAV approach for engineering cultural heritage semantic trajectories as KGs for documentation purposes. In this paper, an extension of our previous work on UAV KG-based semantic trajectories has been presented, contributing (a) an updated methodology for the engineering of STaKGs, (b) an implemented toolset for the management of KG-based semantic trajectories, (c) a refined ontology for the representation of knowledge related to UAVs' semantic trajectories and to cultural heritage documentation, and (d) the application and evaluation of the proposed methodology, the developed toolset, and the ontology, within the domain of UAV-based cultural heritage documentation. In our work, the use of KGs has been promoted in order to be able to (a) integrate external (to UAV datasets) data, enriching the already available flight/mission information, and (b) to make the integrated and enriched data available/open to other services/apps for further reuse (as linked open data). We have demonstrated how our approach is engineered to be able to deliver linked spatiotemporal data (LSD) as an infrastructure for geospatial information systems, advancing this technology towards next-generation spatiotemporal data-driven applications in the UAV and cultural heritage documentation domain.

As already stated, the presented work was originally designed to answer a number of competency questions shaped during the specification phase of STaKG engineering methodology, with the participation of domain experts (geography, geoinformatics, and UAV pilot experts). Table 6 presents this list of competency questions for which the implemented system was successfully evaluated. The related Cypher queries and their sample output results are presented in Appendix C.

Table 6. Competency questions answered by the implemented system.

CQ1 (Appendix C.1)	Which trajectories of a specific mission include records of a specific object?
CQ2 (Appendix C.2)	Which recording positions include records of a specific object?
CQ3 (Appendix C.3)	Which records are produced during a specific mission?
CQ4 (Appendix C.4)	Which missions result in photograph records?
CQ5 (Appendix C.5)	What are the recording positions of a specific flight?
CQ6 (Appendix C.6)	Which records are produced at a specific recording position?
CQ7 (Appendix C.7)	What are the recording segments of a trajectory?
CQ8 (Appendix C.8)	What are the weather conditions at a specific point in time during a specific flight?
CQ9 (Appendix C.9)	Which flights intersect?
CQ10 (Appendix C.10)	What is the number of drones involved in a specific mission and the number of flights initiated for that mission?
CQ11 (Appendix C.11)	What are the recording events that occurred at a distance of less than 100 m from a specific recording event?
CQ12 (Appendix C.12)	Which recording events took place near a specific POI?

Future plans include the extension of the competency questions (Table 6) based on additional user requirements that we plan to obtain from the domain experts. Moreover, our future plans include the extension of analytics supported by our toolset in terms of more sophisticated spatiotemporal analysis on the recorded trajectories, utilizing spatial functions such as "overlap", "touches", and "contains" for geospatial projection trajectories and their parts (currently, our solution supports "intersection" programmatically). Last but not least, we plan to provide a web-based front-end as a free online service for experts that need to follow the proposed STaKG framework pipeline by themselves, i.e., instruction on how to upload a .csv file with a recording mission/flight data of their UAV, and how to obtain access to the generated STaKGs, supported by a free set of visualization and analytics tasks. At the moment, interested readers/users may browse sample data using the implemented toolset from our prototype implementation at http://semantics.aegean.gr (related link for STaKG browser).

Author Contributions: Conceptualization, K.K., S.A., E.M., V.K. and M.V.; methodology, K.K., S.A., E.M., V.K. and M.V.; software, S.A., E.M. and V.K.; validation, E.-E.P. and N.S.; formal analysis, K.K., S.A. and E.M.; investigation, K.K., S.A. and E.M.; resources, S.A., E.-E.P. and N.S.; data curation, S.A. and E.-E.P.; writing—original draft preparation, K.K., S.A. and E.M.; writing—review and editing, K.K., S.A., E.M., V.K., S.A., E.-E.P. and M.V.; visualization, S.A.; supervision, K.K., M.V. and N.S.; project administration, K.K. and M.V.; funding acquisition, M.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Research e-Infrastructure (e-Aegean R&D Network), Code Number MIS 5046494, which is implemented within the framework of the "Regional Excellence" Action of the Operational Program "Competitiveness, Entrepreneurship and Innovation". The action was co-funded by the European Regional Development Fund (ERDF) and the Greek Government (Partnership and Cooperation Agreement 2014–2020).

Data Availability Statement: Samples of the used datasets (in CSV and RDF syntax) and the related ontology for their semantic annotation are accessible online at https://github.com/sotirisAng/stakg and https://github.com/KotisK/onto4drone. In addition, the semantically annotated geographical dataset of the University of the Aegean is available to the public through an SPARQL endpoint at http://semantics.aegean.gr (from SPARQL endpoint link).

Acknowledgments: Authors acknowledge the contribution of Andreas Soularidis, an student at the Dept. of Cultural Technology and Communication, University of the Aegean.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Queries and Sample Responses

Appendix A.1. Cypher Query—ST CreationAppendix

```
MERGE (tr:Trajectory {name:”Traj1”})

MATCH (c:Class {name:”Trajectory”})

MERGE (tr)-[:rdf__type]->(c)

SET tr.name = ’Trajecoty_Name’, tr.label = ‘Traj’+ id(tr), tr.uri = “http://i-lab.aegean.gr/kotis/
ontologies/onto4drone#\T1\textquotedblright+\T1\textquoterightTraj’+ id(tr)

LOAD CSV WITH HEADERS FROM ‘log.csv’ AS line

WITH line

WHERE toInteger(line.pointId)%30 = 0

CREATE (tr)-[hp:hasPart]->(rp:RawPosition)

SET rp.name =‘RawP’+ id(rp), rp.label = ‘RawP’+ id(rp), rp.uri = “http://i-lab.aegean.gr/kotis/
ontologies/onto4drone#\T1\textquotedblright+\T1\textquoterightRawP’+ id(rp), rp.speed =
line[“speed”]

CREATE (p:Point {location:point({longitude:toFloat(line[“GPS:Long”]),
latitude:toFloat(line[“GPS:Lat”])}), longitude:line[“Longitude”], latitude:line[“Latitude”],,
altitude:line[“Altitude”]})

SET p.label = ‘Point’+ id(p), p.name = ‘Point’+ id(p), p.uri = “http://i-lab.aegean.gr/kotis/
ontologies/onto4drone#\T1\textquotedblright+\T1\textquoterightpoint’+ id(p)

CREATE (t:Time {hasTime:line[“GPS:dateTimeStamp”]})
```

```
SET t.label = ‘t’+id(t), t.name = ‘t’+id(t), t.uri = “http://i-lab.aegean.gr/kotis/ontologies/onto4drone#\T1\textquotedblright+\T1\textquoterightt’+ id(t)

create (rp)-[:hasGeometry]->(p)

create (rp)-[:hasTemporalFeature]->(t)
```

Appendix A.2. Cypher Query—ST Enrichment with Record MetadataAppendix

```
LOAD CSV WITH HEADERS FROM ‘records.csv’ AS line
WITH line
CALL {
  WITH line
  MATCH (tr:Trajectory {name:”TrajectoryName”})
  MATCH (rp:RawPosition)
  MATCH (p2:Point)<-[:hasGeometry]-(rp)
  MATCH (rp)-[:hasTemporalFeature]->(t2)
WITH rp, p2, point.distance(point({longitude:toFloat(line[“longitude”]),
latitude:toFloat(line[“latitude”])}), p2.location) as distance,
duration.inSeconds(dateTime(line[“t”]), datetime(t2.hasTime)) as timeDif
RETURN rp
ORDER BY distance, timeDif   asc limit 1 }
SET rp:RecordingPosition
SET rp.name =‘RecP’+ id(rp), rp.label = ‘RecP’+ id(rp)
CREATE (re:RecordingEvent)
SET re.name =‘RecEv’+ id(re), re.label = ‘RecEv’+ id(re), re.uri = “http://i-lab.aegean.gr/kotis/ontologies/onto4drone#\T1\textquotedblright+\T1\textquoterightRecEv’+ id(re)
CREATE (re)-[:occurs]->(rp)
CREATE (record:Record {label:line.title, name:line.title})
CREATE (re)-[:produces]->(record)
```

Appendix A.3. Overpass Query and Sample Response for Fetching Nearby POIs

Request:

```
[out:json];
node(around:50,{longitude},{latitude)-> .poi;
node.poi[name];
out geom;
```

Sample Response:

```
{
  “type”:   “FeatureCollection”,
“features”: [
  {
  “type”: “Feature”,
  "properties": {
      "@id": "node/5389966411",
      "amenity": "university",
      "name": "Τμήμα Γεωγραφίας",
      "toilets:wheelchair": "no",
      "wheelchair": "yes"
  },
  "geometry": {
      "type": "Point",
      "coordinates": [ 26.5692066, 39.0848178 ]
  },
  "id": "node/5389966411"
  }
  ]
}
```

Appendix A.4. GeoSparql Query and Sample Response for Fetching Nearby POIs from UoA SPARQL Endpoint

Query:

```
PREFIX geo: <http://www.opengis.net/ont/geosparql#\T1\textgreater{}
PREFIX geof: <http://www.opengis.net/def/function/geosparql/\T1\textgreater{}
PREFIX uoa: <http://semantics.aegean.gr/ontology/\T1\textgreater{}
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#\T1\textgreater{}
PREFIX units: <http://www.opengis.net/def/uom/OGC/1.0/\T1\textgreater{}
SELECT ?entity ?class ?label ?eLabel
WHERE {
  ?entity a ?class.
OPTIONAL{?entity uoa:hasLabel ?eLabel.}
  ?class rdfs:label ?label.
?entity geo:hasGeometry ?geo .
?geo geo:asWKT ?wkt .
  bind
(geof:distance("<http://www.opengis.net/def/crs/EPSG/0/4326\T1\textgreater{}POINT
(#{lon} #{lat})"^^geo:wktLiteral, ?wkt, units:degree) as ?d)
FILTER (?d < 0.0005)
  FILTER contains(str(?wkt),"POINT")
}
```

Sample response:

```
    {“poi”: { “type”: “uri” , “value”:
“http://semantics.aegean.gr/resources/PetrifiedTrunk_3b243cdd74c34c2fBb970a80e8038baf” } ,
    “class”: { “type”: “uri” , “value”: “http://semantics.aegean.gr/ontology/PetrifiedTrunk” },
    “label”: { “type”: “literal” , “value”: “Petrified Trunk” } } ,
    {“poi”: { “type”: “uri” , “value”:
“http://semantics.aegean.gr/resources/PetrifiedTrunk_C26b38420cd0422c900dE3eb91dbe9eb” } ,
    “class”: { “type”: “uri” , “value”: “http://semantics.aegean.gr/ontology/PetrifiedTrunk” },
    “label”: { “type”: “literal” , “value”: “Petrified Trunk” } } ,
    ]
  }
}
```

Appendix B. Visualization and Analytics Queries

Appendix B.1. Trajectory Visualization Query

```
MATCH (tr:Trajectory {name:”UoA1”})
MATCH (tr)-[:hasPart]-(rawp:RawPosition)-[:hasGeometry]-(point:Point)
RETURN point
```

Appendix B.2. Trajectory and the Recording Positions Query

```
MATCH (tr:Trajectory
{name:$neodash_trajectory_name})-[:hasPart]-(rp:RecordingPosition)-[:hasGeometry]-(p:Point)
WITH collect({id: rp.name, label:”Rec”, point: p.location, explore: p.label }) AS rec_points, tr
MATCH (tr)-[:hasPart]-(rawp:RawPosition)-[:hasGeometry]-(n:Point)
WITH collect({id: rawp.name, label:”Raw”, point: n.location}) as raw_points, rec_points
RETURN raw_points, rec_points
```

Appendix B.3. Recording Event Relations

```
MATCH (rec_position:RecordingPosition)-[has_geom:hasGeometry]-(rp:Point)
MATCH (tr:Trajectory)-[hp:hasPart]-(recp)
MATCH (rec_position)<-[oc:occurs]-(rev:RecordingEvent)
MATCH (rev)-[prod:produces]-(record:Record)-[recs:records]-> (poi:PointOfInterest)
WHERE rp.label = $neodash_point_label
RETURN tr,hp, rec_position, has_geom, rp, rev, record, prod, oc, recs, poi
```

Appendix B.4. Recording Segments of Trajectory

```
MATCH(tr:Trajectory{name:$neodash_trajectory_name})-[:hasPart]-(rp:RawPosition)-
[:hasGeometry]-(p:Point)
WHERE (rp)-[:comprises]-(:RecordingSegment)
WITH collect({id: rp.name, label:”Rec”, point: p.location, explore: p.label}) AS rec_points, tr
MATCH (tr)-[:hasPart]-(rawp:RawPosition)-[:hasGeometry]-(n:Point)
WITH collect({id: rawp.name, label:”Raw”, point: n.location}) as raw_points, rec_points
RETURN raw_points, rec_points
```

Appendix B.5. Intersecting Trajectories

```
MATCH (tr:Trajectory {name:”UoA1”})-[:intersects]-(tr2:Trajectory)
MATCH (tr)-[:hasPart]-(rawp:RawPosition)-[:hasGeometry]-(n:Point)
MATCH (tr2)-[:hasPart]-(rawp2:RawPosition)-[:hasGeometry]-(n2:Point)
RETURN {id: rawp.name, label:”traj1”, point: n.location}, {id: rawp2.name, label:”traj2”, point: n2.location}
```

Appendix B.6. Number of Recording Points of a Trajectory

```
MATCH (tr:Trajectory {name:”SigriTraj”}) -
[:hasPart]->(:RecordingPosition)-[:hasGeometry]-(p:Point)
RETURN count(p) as number_of_recording_points
```

Sample Response

```
number_of_recording_points
58
```

Appendix B.7. POIs in a Trajectory

```
MATCH (tr:Trajectory {uri:”http://i-lab.aegean.gr/kotis/ontologies/onto4drone#Traj755”})
MATCH (tr)-[:hasPart]-(:RecordingPosition)-[:occurs]-(:RecordingEvent)-[:produces]-(:Record)-[:records]-(poi:PointOfInterest)
RETURN distinct poi.name as name, poi.uri as uri
```

Sample Response

Name	Uri
“University of the Aegean”	“http://semantics.aegean.gr/resources/University_2585”
“Τμήμα Γεωγραφίας” (Dept. of Geography)	“https://www.openstreetmap.org/node/5389966411”
“Φοιτητική λέσχη” (Students Club)	“https://www.openstreetmap.org/node/5389966410”
“Τμήμα Ωκεανογραφίας και Θαλασσών Βιοεπιστημών” (Dept. of Marine Sciences)	“https://www.openstreetmap.org/node/5389966409”
“Λόφος Ξενία” (University Hill)	“https://www.openstreetmap.org/node/2360266377”

Appendix B.8. Mean Temperature Recorded during a Trajectory

```
MATCH (tr:Trajectory {uri:”http://i-lab.aegean.gr/kotis/ontologies/onto4drone#Traj755”})
MATCH (tr)-[:hasPart]-(:RecordingPosition)-[:occurs]-(:RecordingEvent)-[:produces]-(:Record)-[:records]-(poi:PointOfInterest)
RETURN distinct poi.name as name, poi.uri as uri
```

Sample Response

avg_temperature

16.1

Appendix B.9. Number of POIs That Are of a Specific Type in a Radius R of a Selected Point

```
MATCH (point_a:Point {latitude:"39.208266", longitude:"25.901183"})
MATCH (point_b:PointOfInterest)
WITH point_b, point.distance(point_a.location, point_b.location) AS dist
WHERE dist < 30
RETURN count(point_b) AS num_of_pois
```

Sample Response

num_of_pois
3

Appendix B.10. POIs That Two or More Trajectories Have in Common

```
MATCH (poi:PointOfInterest)- [:records]-(record)<-[:produces]-(:RecordingEvent)-[:occurs]->(:RecordingPosition)<-[:hasPart]-(trajectory:Trajectory)
WITH count(distinct trajectory) AS num_of_trajectories, poi
WHERE num_of_trajectories > 1
RETURN poi.uri
```

Sample Response

poi.uri
"https://www.openstreetmap.org/node/5389966410"
"https://www.openstreetmap.org/node/5389966411"
"https://www.openstreetmap.org/node/2360266377"
"http://semantics.aegean.gr/resources/University_2585"

Appendix C. Competency Question Queries

Appendix C.1. CQ1

```
MATCH (poi:PointOfInterest {name:"KormosIstamenos30"})
MATCH (record:Record)-[:records]-(poi)
MATCH (record)<-[:produces]-(:RecordingEvent)-[:occurs]->
(:RecordingPosition)<-[:hasPart]-(trajectory:Trajectory)
RETURN DISTINCT trajectory.name
```

Sample Response

trajectory.name: "SigriTraj"

Appendix C.2. CQ2

```
MATCH (poi:PointOfInterest {name:”KormosIstamenos30”})
MATCH (record:Record)-[:records]-(poi)
MATCH (record)<-[:produces]-(:RecordingEvent)-[:occurs]->(:RecordingPosition)<-[:hasPart]-
(trajectory:Trajectory)
MATCH (trajectory)<-[:formsTrajectory]-(:Flight)<-[:includesFlight]-(mission:Mission)
WITH mission, poi
MATCH (mission)-[:includesFlight]->(:Flight)-[:formsTrajectory]->
(tr:Trajectory)-[:hasPart]-(rec_pos:RecordingPosition)
RETURN DISTINCT rec_pos.uri as uris
```

Sample Response

```
URIs
“http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RawP4545”
“http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RawP4554”
“http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RawP4588”
“http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RawP4544”
“http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RawP4609”
“http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RawP4576“
```

Appendix C.3. CQ3

```
MATCH (mission:Mission {name:”UoAMission1”})
MATCH (mission)-[:includesFlight]->(:Flight)-[:formsTrajectory]->(tr:Trajectory)-[:hasPart]-
(rec_pos:RecordingPosition)-[:occurs]-(rec_event:RecordingEvent)
MATCH (rec_event)-[:produces]-(record:Record)
RETURN DISTINCT record.name as record_names
```

Sample Response

```
record_names
“DJI_0007.JPG”
“DJI_0006.JPG”
“DJI_0003.JPG”
“DJI_0018.JPG”
“DJI_0005.JPG”
“DJI_0002.JPG”
```

Appendix C.4. CQ4

```
MATCH (m:Mission)-[:includesFlight]-> (fl:Flight)-[:formsTrajectory]->(:Trajectory)-[:hasPart]-
>(:RecordingPosition)<-[:occurs]-(:RecordingEvent)-[:produces]->(ph:Photograph)
RETURN m.name AS mission_name
```

Sample Response

```
mission_name
“SigriMission1”
“UoAMission1”
```

Appendix C.5. CQ5

```
MATCH (fl:Flight {name:"UoAFlight1"})-[:formsTrajectory]->(:Trajectory)-[:hasPart]->(rp:RecordingPosition)-[:hasGeometry]-(p:Point)
RETURN p.latitude as latitude, p.longitude as longitude
```

Sample Response

```
"39.208266" "25.901183"
"39.208408" "25.901207"
"39.208146" "25.90103"
"39.208257" "25.901175"
"39.208003" "25.901185"
"39.208006" "25.901119"
```

Appendix C.6. CQ6

```
MATCH (rec:Record)-[:produces]-(:RecordingEvent)-[:occurs]-(:RecordingPosition {name: "RecP880"})
RETURN rec.name
```

Sample Response

```
name: "DJI_0051.JPG"
```

Appendix C.7. CQ7

```
MATCH (rec:Record {name: "DJI_0051.JPG"})-[:produces]-(:RecordingEvent)-[:occurs]-(:RecordingPosition)<-[:comprises]-(rseg:RecordingSegment)
RETURN rseg.uri
```

Sample Response

```
reseg.uri: "http://i-lab.aegean.gr/kotis/ontologies/onto4drone#rs2045"
```

Appendix C.8. CQ8

```
MATCH (fl:flight {name: "uoaflight1"})
MATCH (p:Point {latitude: "39.0850988", longitude: "26.569415"})
MATCH (fl)-[:formsTrajectory]->(:Trajectory)-[:hasPart]-(rec_pos)-[:hasGeometry]-(p)
MATCH (rec_pos)-[:hasWeatherCondition]-(wc:WeatherCondition)
RETURN wc
```

Sample Response

```
{
  "identity": 5370,
  "labels": [
      "WeatherCondition"
  ],
  “properties”: {
“windSpeedMax”: 16.3,
“reportedPressure”: 993.8,
“reportedMaxTemperature”: 13.1
  }
}
```

Appendix C.9. CQ9

```
MATCH (fl:Flight {name:”UoAFlight1”})-[:formsTrajectory]-(:Trajectory)-
[:intersects]->(:Trajectory)-[:formsTrajectory]-(fl2:Flight)
RETURN fl2.name AS intersecting_flight
```

Sample Response

```
intersecting_flight
”UoAFlight2”
```

Appendix C.10. CQ10

```
MATCH (m:Mission {name:”UoAMission1”})-[:includesFlight]->(fl:Flight)
MATCH (dr:UAVDrone)-[:hasFlight]->(fl)
RETURN count(fl) as num_of_flights, count(dr) as num_of_drones
```

Sample Response

```
num_of_flights num_of_drones
2                2
```

Appendix C.11. CQ11

```
MATCH (:RecordingEvent {name: “RecEv5272”})-[:occurs]->(:RecordingPosition)-
[:hasGeometry]-(point_a:Point)
WITH point_a.location as pointA
MATCH (point_b:Point)
WITH point_b.location as pointB, pointA
WHERE point.distance(pointA, pointB) < 100
MATCH (p:Point
{location:pointB})<-[:hasGeometry]-(:RecordingPosition)-[:occurs]-(rec_ev:RecordingEvent)
RETURN rec_ev.label
```

Sample Response

```
rec_ev.label
"RecEv5261"
"RecEv5233"
"RecEv5223"
"RecEv5237"
```

Appendix C.12. CQ12

```
MATCH (poi:PointOfInterest {name:"KormosIstamenos30"})
WITH poi.location as pointA
MATCH (point_b:Point)
WITH point_b.location as pointB, pointA
WHERE point.distance(pointA, pointB) < 100
MATCH (p:Point
{location:pointB})<-[:hasGeometry]-(:RecordingPosition)-[:occurs]-(rec_ev:RecordingEvent)
RETURN rec_ev.uri
```

Sample Response

```
rec_ev.uri
"http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RecEv5261"
"http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RecEv5233"
"http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RecEv5223"
"http://i-lab.aegean.gr/kotis/ontologies/onto4drone#RecEv5237"
```

References

1. Parent, C.; Spaccapietra, S.; Renso, C.; Andrienko, G.; Andrienko, N.; Bogorny, V.; Damiani, M.L.; Gkoulalas-Divanis, A.; Macedo, J.; Pelekis, N.; et al. Semantic trajectories modeling and analysis. *ACM Comput. Surv.* **2013**, *45*, 1–32. [CrossRef]
2. Santipantakis, G.; Vouros, G.; Glenis, A.; Doulkeridis, C.; Vlachou, A. The datAcron Ontology for Semantic Trajectories. In *Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer International Publishing: Berlin/Heidelberg, Germany, 2017; Volume 10577, pp. 26–30. [CrossRef]
3. Ehrlinger, L.; Wöß, W. Towards a definition of knowledge graphs. In Proceedings of the 12th International Conference on Semantic Systems (SEMANTICS2016): Posters and Demos Track, CEUR Workshop Proceedings, Leipzig, Germany, 12–15 September 2016; Volume 1695.
4. Hogan, A.; Blomqvist, E.; Cochez, M.; D'Amato, C.; De Melo, G.; Gutierrez, C.; Kirrane, S.; Gayo, J.E.L.; Navigli, R.; Neumaier, S.; et al. Knowledge Graphs. *ACM Comput. Surv.* **2021**, *54*, 1–37. [CrossRef]
5. Carriero, V.A.; Gangemi, A.; Mancinelli, M.L.; Nuzzolese, A.G.; Presutti, V.; Veninata, C. Pattern-based design applied to cultural heritage knowledge graphs. *Semant. Web* **2021**, *12*, 313–357. [CrossRef]
6. Fensel, D.; Şimşek, U.; Angele, K.; Huaman, E.; Kärle, E.; Panasiuk, O.; Toma, I.; Umbrich, J.; Wahler, A. Introduction: What Is a Knowledge Graph? *Knowl. Graphs* **2020**, 1–10. [CrossRef]
7. Gao, C.; Zhang, Z.; Huang, C.; Yin, H.; Yang, Q.; Shao, J. Semantic trajectory representation and retrieval via hierarchical embedding. *Inf. Sci.* **2020**, *538*, 176–192. [CrossRef]
8. Moraitou, E.; Angelis, S.; Kotis, K.; Caridakis, G.; Papadopoulou, E.E.; Soulakellis, N. Towards Engineering Drones Semantic Trajectories as Knowledge graphs. In Proceedings of the 5th International Workshop on Geospatial Linked Data (GeoLD 2022), Co-Located with the 19th European Semantic Web Conference (ESWC 2022), Crete, Greece, 29 May–2 June 2022.
9. Pinto, H.S.; Staab, S.; Tempich, C. DILIGENT: Towards a fine-grained methodology for DIstributed, Loosely-controlled and evolvInG Engineering of oNTologies. *Front. Artif. Intell. Appl.* **2004**, *110*, 393.
10. Kotis, K.; Vouros, G.A. Human-centered ontology engineering: The HCOME methodology. *Knowl. Inf. Syst.* **2005**, *10*, 109–131. [CrossRef]
11. How-to: Building Knowledge Graphs in 10 Steps | Ontotext Fundamentals. Available online: https://www.ontotext.com/knowledgehub/fundamentals/how-to-building-knowledge-graphs-in-10-steps/ (accessed on 25 November 2022).
12. datAcron Ontology. Available online: http://ai-group.ds.unipi.gr/datacron_ontology/ (accessed on 25 November 2022).
13. dronetology, the UAV Ontology. Available online: http://www.dronetology.net/ (accessed on 25 November 2022).

14. W3C Geospatial Ontologies. Available online: https://www.w3.org/2005/Incubator/geo/XGR-geo-ont-20071023/ (accessed on 25 November 2022).
15. Linked Open Vocabularies. Available online: https://lov.linkeddata.es/dataset/lov (accessed on 25 November 2022).
16. Ontology Design Patterns. org (ODP)—Odp. Available online: http://ontologydesignpatterns.org/wiki/Main_Page (accessed on 25 November 2022).
17. DOLCE+DnS Ultralite Ontology. Available online: http://www.ontologydesignpatterns.org/ont/dul/DUL.owl (accessed on 25 November 2022).
18. SKOS Simple Knowledge Organization System Reference. Available online: https://www.w3.org/TR/skos-reference/ (accessed on 25 November 2022).
19. Semantic Sensor Network Ontology. Available online: https://www.w3.org/ns/ssn/ (accessed on 25 November 2022).
20. Simple Features Geometries. Available online: http://schemas.opengis.net/sf/1.0/simple_features_geometries.rdf (accessed on 25 November 2022).
21. Simple Feature Access—Part 1: Common Architecture | OGC. Available online: https://www.ogc.org/standards/sfa (accessed on 25 November 2022).
22. GML Geometries. Available online: http://schemas.opengis.net/gml/3.2.1/gml_32_geometries.rdf (accessed on 25 November 2022).
23. OGC GeoSPARQL 1.0. Available online: http://schemas.opengis.net/geosparql/1.0/geosparql_vocab_all.rdf (accessed on 25 November 2022).
24. UNESCO Thesaurus. Available online: https://vocabularies.unesco.org/browser/thesaurus/en/ (accessed on 25 November 2022).
25. EKT-UNESCO Thesaurus. Available online: https://www.semantics.gr/authorities/vocabularies/ekt-unesco/vocabulary-entries (accessed on 25 November 2022).
26. GitHub—KotisK/Onto4drone: An Ontology for Representing Knowledge Related to Drones and Their Semantic Trajectories. Available online: https://github.com/KotisK/onto4drone (accessed on 25 November 2022).
27. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. Available online: https://www.w3.org/Submission/SWRL/ (accessed on 25 November 2022).
28. Shapes Constraint Language (SHACL). Available online: https://www.w3.org/TR/shacl/ (accessed on 25 November 2022).
29. SPARQL Query Language for RDF. Available online: https://www.w3.org/TR/rdf-sparql-query/ (accessed on 25 November 2022).
30. OpenRefine. Available online: https://openrefine.org/ (accessed on 25 November 2022).
31. Karma: A Data Integration Tool. Available online: https://usc-isi-i2.github.io/karma/ (accessed on 25 November 2022).
32. RDF Triple Stores vs. Labeled Property Graphs: What's the Difference? Available online: https://neo4j.com/blog/rdf-triple-store-vs-labeled-property-graph-difference/ (accessed on 25 November 2022).
33. Graph Data Platform | Graph Database Management System | Neo4j. Available online: https://neo4j.com/ (accessed on 25 November 2022).
34. Kyzirakos, K.; Karpathiotakis, M.; Koubarakis, M. Strabon: A Semantic Geospatial DBMS. Available online: http://www.strabon.di.uoa.gr (accessed on 25 November 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.