LandSense

A Citizen Observatory and Innovation Marketplace for Land Use and Land Cover Monitoring

Deliverable 5.3

Adaptation measures of COBWEB quality assurance service for the LandSense Citizen Observatory

European Commission

Horizon 2020
European Union funding
for Research & Innovation

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.689812

Project acronym:	LandSense
Project title:	A Citizen Observatory and Innovation Marketplace for Land Use and Land Cover Monitoring
Project number:	689812
Instrument:	Horizon 2020
Call identifier:	SC5-17-2015
Topic	Demonstrating the concept of citizen observatories
Type of action	Innovation action

Start date of project:	01-09-2016
Duration:	48 months

Deliverable number	**D5.3**
Deliverable title	**Adaptation measures of COBWEB quality assurance service for the LandSense Citizen Observatory**
Deliverable due date	**30-11-2018**
Lead beneficiary	**HELYX**
Work package	**5**
Deliverable type	**Report**
Submission date: **Revision:**	**04-01-2019** **Version 1.0**

Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium	
CO	Confidential, only for members of the consortium (including the Commission Services)	X

Title:
LandSense Quality Assurance Services
Author(s)/Organisation(s):
Sam Meek/Helyx SIS, Matthew Knight/Helyx SIS, Theo Brown/Helyx SIS
Contributor(s):
Julian Rosser/University of Nottingham

Short Description:
This report outlines the development of the LandSense Quality Assurance Services. The services take inspiration from the EU funded Citizen Observatory WEB (COBWEB) project and re-use some of the implemented procedures. COBWEB utilised an Open Geospatial Consortium Web Processing Service (WPS), the standard for exposing web based geoprocessing services. Although still in use, Representational State Transfer (REST) interfaces have largely supplanted traditional web services, therefore the LandSense quality assurance services are exposed via a standardised REST API. The baseline standard for data quality in geographic information is ISO 19157, therefore, some of the tests implemented are taken directly from this standard. Since the conclusion of COBWEB there has been an explosion in Artificial Intelligence (AI) procedures, specifically for classifying the content of imagery. The LandSense quality assurance services make use of AI for facial recognition (to aid GDPR compliance) and photograph content analysis for crop recognition (linked to LandSense demo case study needs). Facial recognition is a well-known AI procedure and is implemented using out of the box technology, crop recognition is more experimental, therefore, further work has been done to test the quality of the developed AI models. The services developed in this work package have been tailored to the needs of the LandSense project stakeholders.
Keywords:
Quality Assurance; Artificial Intelligence; Standards; Web Services

History:				
Version	Author(s)	Status	Comment	Date
0.1	Sam Meek	Draft	Draft with integrated inputs from contributors	14/12/2018
0.2	Sam Meek	Final Draft	Edits from review	18/12/2018
1.0	Sam Meek	Submitted version	Final edits and formatting	04/01/2019

Review:			
Version	Reviewer	Comment	Date
0.1	Giles Foody	Content review	16/12/2018
0.2	Inian Moorthy	Review of content, figures, tables. Approved for submission	04/01/2019

Table of Contents

List of Figures

List of Tables

1 Introduction

This document describes the LandSense Quality Assurance (QA) services that are deployed on the LandSense Engagement Platform (LEP - https://lep.landsense.eu/). The motivating work for deployment of the services was carried out in the European Union Funded FP7 Citizen Observatory WEB (COBWEB) project that completed in 2016. The COBWEB project attempted to design and develop a set of quality assurance processes based upon those described in ISO19157 and the flexible framework for quality assuring crowdsourced data explored in Meek *et al* (2014). This document outlines the LandSense QA services, provides rationale for the changes made to the COBWEB platform, lists algorithmic tests implemented from ISO 19157 and outlines the work done using Artificial Intelligence (AI) techniques to perform image content recognition in relation to LandSense applications.

1.1 Citizen Observatory WEB

In some respects, the LandSense project is the spiritual successor to the COBWEB project as it aims to re-use some of the work done in COBWEB WP4, the quality assurance work package and continues the important citizen's observatories work. COBWEB was focused on the qualification of data collected by citizen scientists across the three use cases of:

- Earth Observation product enhancement.
- Flooding.
- Biological monitoring.

LandSense is focused on Earth Observation data, therefore, some of the work done in COBWEB is directly transferrable to the LandSense Qualification Services as highlighted in deliverable 5.1. A difference between COBWEB and LandSense is that COBWEB aimed to provide an end-to-end experience by designing and developing mobile applications, middleware and backend service. LandSense instead aims to provide a platform for others to engage with by building their own clients to interact with the services.

The technological assets from COBWEB include the following:

- An Open Geospatial Consortium (OGC) compliant Web Processing Service (WPS) containing executable procedures conformant to ISO 19157 and Meek et al (2014).
- A Business Process Modelling Notation (BPMN) workflow engine that enables authorised users to orchestrate and configure workflows (Meek *et al* 2016).

The findings from COBWEB suggested that the qualification aspects of the system were useful in a set of use cases, but the orchestration aspects of the system were confusing, even for project administrators. Additionally, there is a compounding issue of intra-workflow interoperability of processes (i.e. appropriate process chaining was difficult to communicate due to the possible failure of workflows due to process null results). Figure 1 shows the generic processes of COBWEB as well as a workflow for qualifying Ash Dieback, a disease prevalent in the UK.

Due to the complexity of workflow orchestration in these use cases, these aspects of the system have been deprecated in LandSense and the focus has returned to production of quality assurance services that are suited to the LandSense use cases.

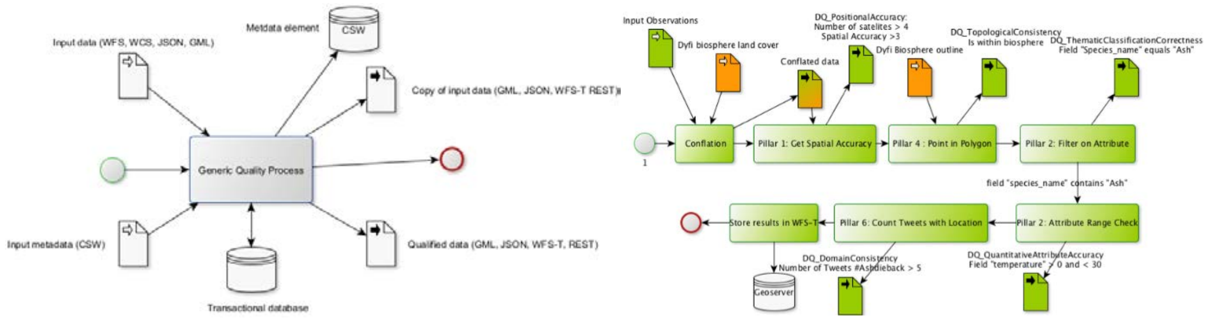

Figure 1 Left: Generic WPS process model. Right: Example COBWEB Ash Dieback workflow.

2 Component design

The LandSense Quality Assurance services utilise the spirit of COBWEB procedures, but they have been re-implemented in a new framework using a modern paradigm. A Representational State Transfer (REST) based Application Programming Interface (API) using OpenAPI as the design pattern is utilised for the following reasons:

- REST is resource based rather than service based, which means that it is possible to expose endpoints with different purposes. In the COBWEB version of QA, we were restricted to a Web Processing Service (WPS), which is specifically designed for processing.
- REST interfaces are modern utilised in many different use cases across the web. Additionally many clients work *out-of-the-box* with REST endpoints making development and testing of clients simple.
- OpenAPI is growing in popularity and looks to be adopted by the OGC in their Web Feature Service (WFS) version 3.0, with other relevant OGC services likely to follow suit.
- OpenAPI is flexible and processes to support a variety of different back end services are easily implementable. In WPS, there is no standard was to *adapt* to existing systems (although there is a small push within the OGC to provide this as an extension), therefore every interface written to accommodate an external system is bespoke (see Schumann et al 2018). The OpenAPI specification assumes that it will be used to expose a variety of procedures and services and is therefore simple to do so.

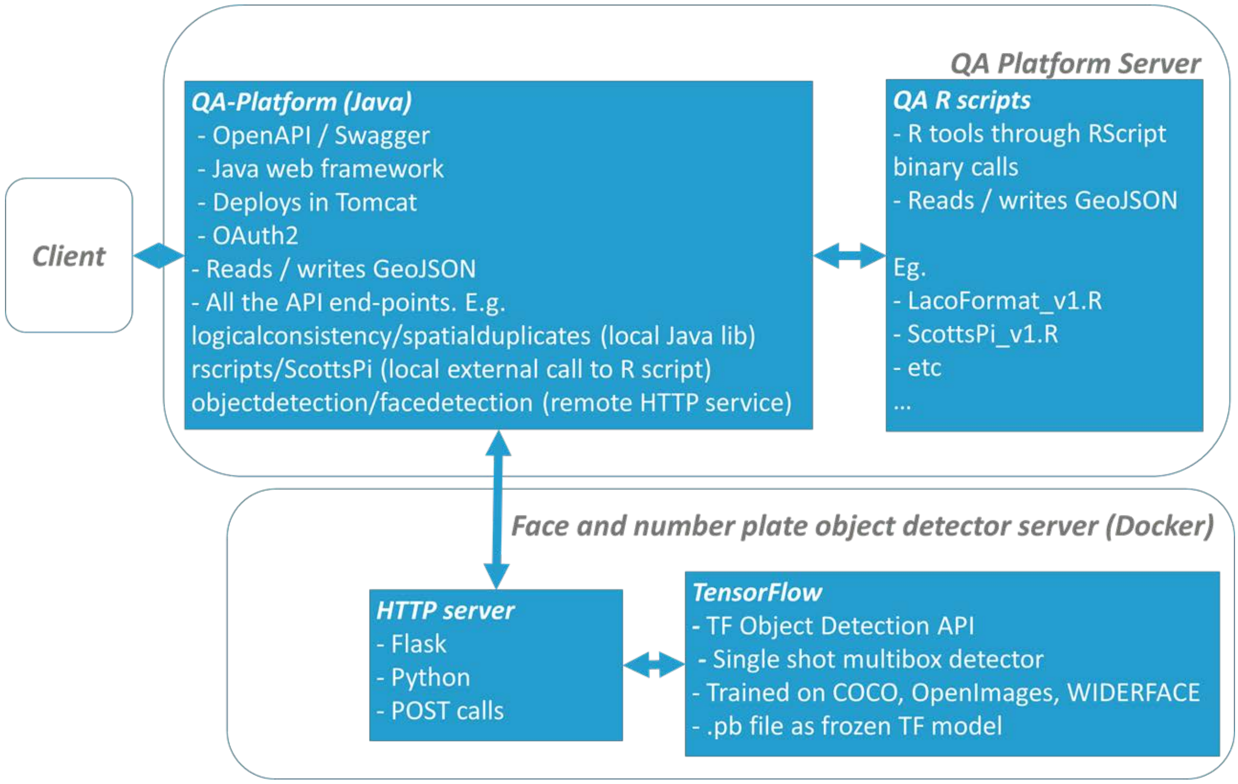

Figure 2 Generic API interaction design

Figure 2 shows the generic design of the LandSense quality assurance services. The QA services sit on the *LandSense Engagement Platform* and are made available to authorised clients through a generic REST api. The QA Platform Server contains procedures directly implemented in *R* (procedures described in D 5.2) and Java and the *QA-Platform* exposes the Docker based facial recognition and Crop Modelling procedures. The exact nature of these approaches is opaque to the user, who just sees a REST endpoint.

2.1 OpenAPI and REST

OpenAPI[i] is a forward looking specification created by a consortium of experts who recognise the value of standardising REST API descriptions. In LandSense, it is used as a method of exposing algorithms, services and resources in an interoperable and reusable way. OpenAPI uses standard Hyper Text Transfer Protocol (HTTP) verbs to enable interaction with the resources:

- GET
- PUT
- POST
- DELETE
- HEAD

The verbs have different uses depending on the endpoint, but in general GETs are used to get a resource, PUTs are for transactions, POST is to send a list of instructions, DELETE does the opposite of PUT and HEAD is the same as the GET request, but just retrieves the header and does not request the body. These verbs are well-known and a common way to guide interaction between a client and server.

The driving force behind a standardised interface is the ability of implementers to build clients for the services quickly and easily. LandSense as a project does not intend to provide clients for all use cases, but instead to provide guidance on how they should be implemented. Figure 3 shows the OpenAPI/Swagger interface used for testing. Note that the endpoint locations follow the ISO 19157 descriptions of tests (for example /logicalconsistency/).

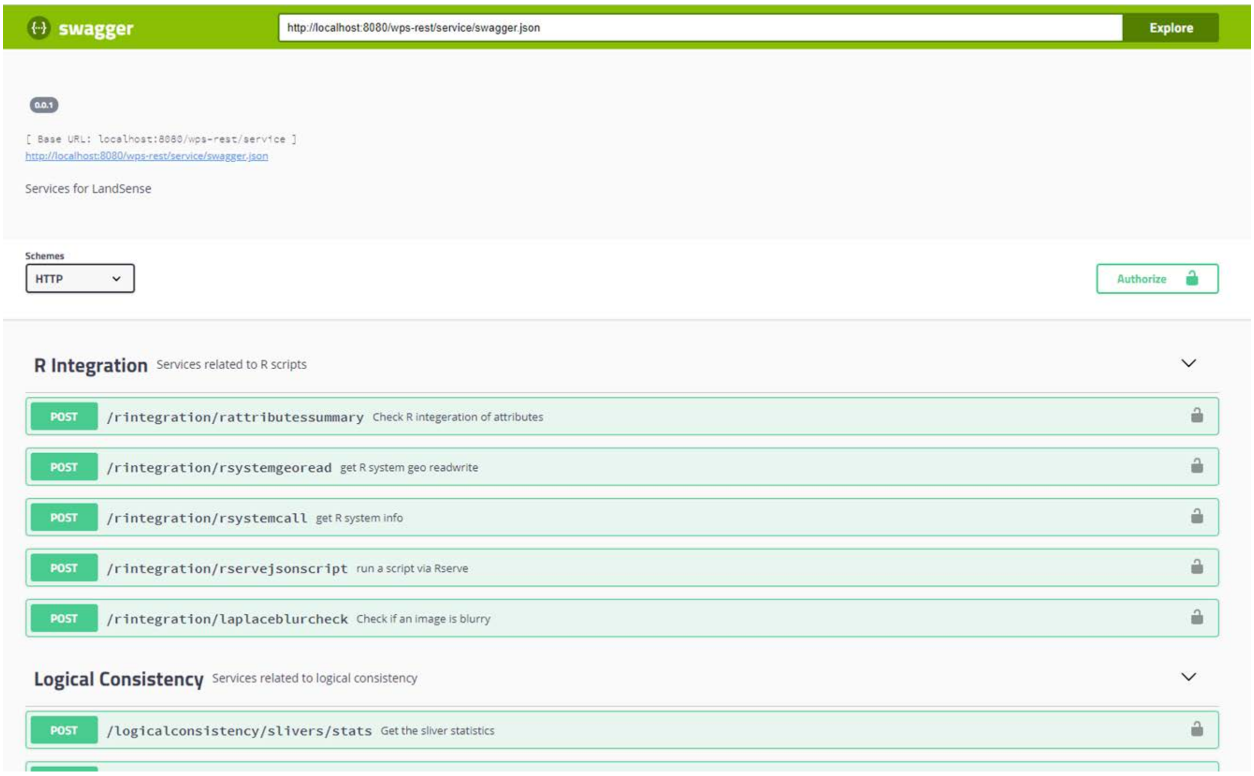

Figure 3 Swagger interface

2.2 ISO 19157 tests

ISO 19157 – Geographic Information: Data Quality is a foundational document that is seen as an important starting point for many data quality endeavours. In LandSense, the tests available were considered in turn and appropriate tests for the LandSense use cases were implemented and exposed via the OpenAPI interface. Table 1 describes the tests implemented so far from the ISO 19157 standard document. The standard was originally written in 2003 as ISO 19115 and has had several revisions, however its concepts of data quality were defined before AI and machine learning techniques were a viable option to use operationally. Therefore, many of the tests are unsuitable in their described form to establish data quality in citizen science style use cases, although most of the concepts are generic enough to be sound.

Although the ISO 19157 tests described in the document may not be directly attributable to qualification of crowdsourced data, the concepts still largely appear to be valid. For example *Thematic Accuracy – Classification Correctness* is realised by calculating the number of features miss-classified into a class, but does not provide guidance on how one might calculate whether a data point is correctly classified or not unless there is a manual undertaking to match tags with image content. New AI techniques that can perform image content classification appear to be able to at least partially fill that gap.

Table 1 List of tests implemented from ISO 19157

Table 1 ISO 19157 tests available			
ISO Table	**Super-class**	**Sub-class**	**Notes**
D11	Logical Consistency	Conceptual Consistency	Number of overlaps for a polygon
D14	Logical Consistency	Domain Consistency	Value domain non-conformance
D16	Logical Consistency	Domain Consistency	Dataset level number of non-conforming items
D17	Logical Consistency	Domain Consistency	Dataset level conformance rate
D18	Logical Consistency	Domain Consistency	Dataset level non-conformance rate
D29	Positional Accuracy	Absolute positional accuracy	Mean value of positional uncertainties
D63	Thematic Accuracy	Classification correctness	Number of incorrectly classified features
D64	Thematic Accuracy	Classification correctness	Rate of D63

As mentioned previously, the LandSense services are provided as a REST api and it is up to the users of the service to implement clients as they see fit. A demonstrator client was implemented on the LEP and simply qualified numbers of overlapping polygons (Figure 4). Figure 4 shows an example of other ISO 19157 tests including completeness, topological consistency and accuracy. This was not implemented as a client in LandSense, but gives an example of outputs that users would see should they use these services.

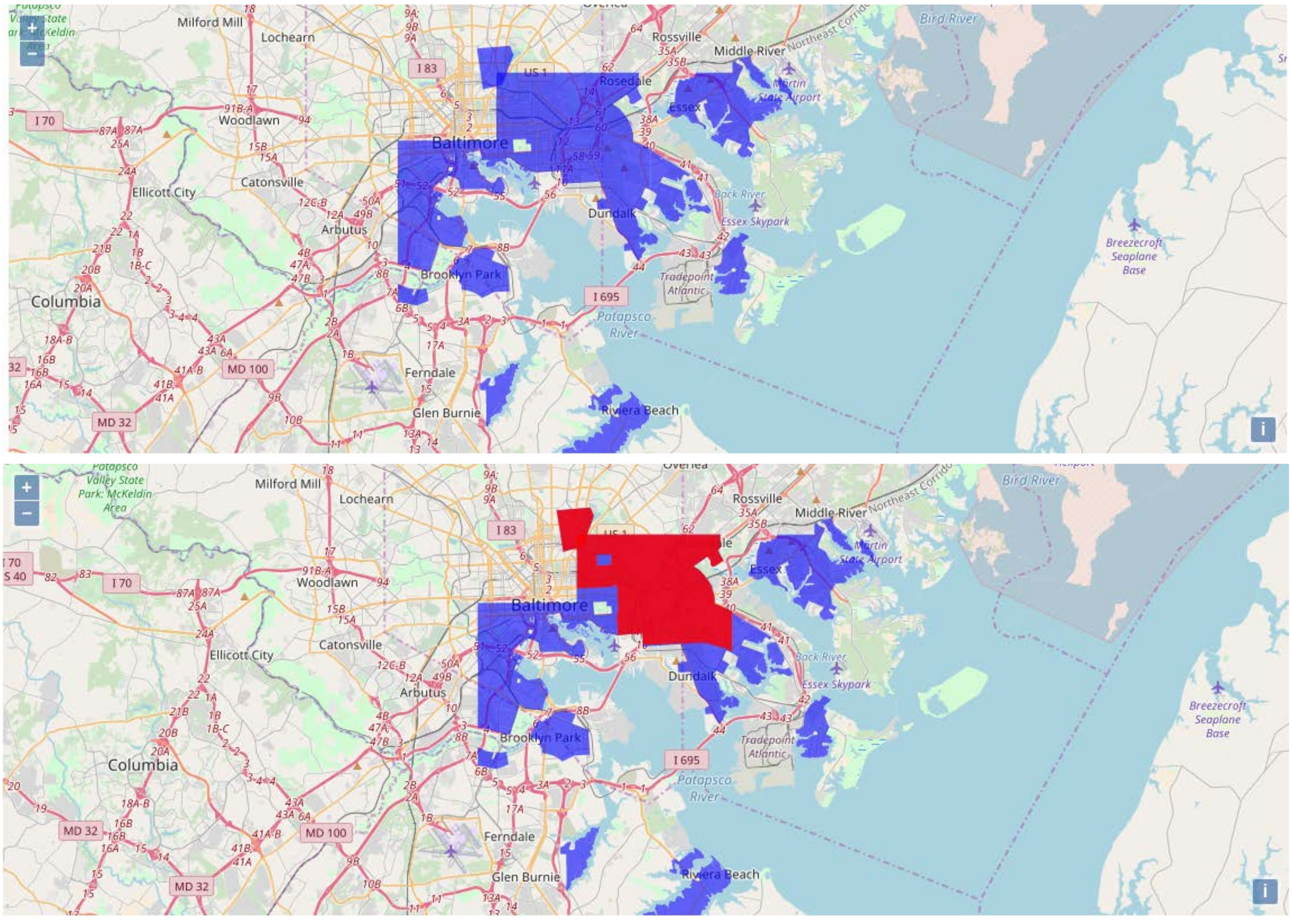

Figure 4 Example client determining overlapping polygons

2.2.1 Photograph blur

Although not technically part of the ISO suite of tests, a Laplace transform was employed to check for photograph blur. This was done to provide instant feedback to the user that their submission should be retaken as it is unsuitable for consideration. Laplace transform identifies blurriness in photographs by performing a series of procedures on an image to compare *steepness* between edges on an image. Although not an *exact* measurement of blurriness, the technique provides a good proxy. The full procedure is as follows:

- Convert image to grey scale and resize.
- Perform a Laplace transform using a 3 x 3 kernel i.e [0,-1,0; -1,4,-1; 0,-1,0].
- Calculate the variance of the transformed image against a given threshold (e.g. 1500)

The user provides a threshold input value corresponding to the variance to indicate the level of blurriness that they are willing to accept. Higher values correspond to clearer image. It is envisaged that this procedure be used as part of a set of processes such as AI to determine the overall suitability of a user submitted photograph for use.

2.3 Security

Access to the LandSense QA services is protected by OAuth2 authorisation services provided by Secure Dimensions and the WP3 team. In order to access the services, the user logs into the federation and authorises the LandSense QA services application. They then receive an access token that can be used in a request to authorise the use of the LandSense QA services. If accessing the services by a remote client, then the token is passed in the request header. The Swagger document also contains the ability to authenticate the user by manually entering the *bearer token* (Figure 5).

Figure 5 Authentication in the Swagger UI

Further explanation of the security model utilised in LandSense can be found in the **WP3 – Build** work package deliverables, specifically D3.1 – LandSense Citizen Observatory Reference Framework, D3.2 – Spatial Data Infrastructure and data ingestion workflows, and D3.3 LandSense Engagement Platform.

2.4 Artificial Intelligence & Docker

A big addition to the LandSense data quality services over the COBWEB is the implementation of services that utilise AI to produce a classification of the contents of an image. This has been done in two different contexts:

1) Identifying potential aspects of the image that might breach GDPR rules, specifically faces.
2) Automatically identifying plants that are the subject of a crowdsourced image.

These technology components are deployed in a container and exposed via the OpenAPI interface described above. A key advantage of Docker is the efficiency and ease of exposing the functionality via the API, therefore it is likely that all future enhancements to the LandSense QA services will be done via deployment of Docker containers.

The AI engine of choice is TensorFlow (tensorflow.org) due to vendor familiarity and general acceptance in the industry. It utilises deep learning neural networks to produce models for machine learning. TensorFlow computations are expressed as stateful dataflow graphs. The operations on performed on the multidimensional arrays are known as *tensors,* hence the name of the software. TensorFlow was originally built by Google employees for internal use but has since been released on Apache 2 licence for open source use.

2.5 AI for facial recognition

Facial recognition is a typical use of AI and is utilised in a multitude of applications. This section briefly describes the implementation for facial recognition in LandSense.

The Object Detection model was created using TensorFlow and Google's TensorFlow Object Detection API. The high level steps of creating such a model are as follows:

- Prepare data to ensure each image has appropriately labelled objects.
- Convert object labels to TensorFlow Records and create label mapping.
- Choose a pre-trained model, either one from the TensorFlow API or a custom one, and configure the model for training. In this case the model used was mobilenet SSD (single shot multibox detector) included in the TensorFlow API.
- Run TensorFlow Object Detection API's training function to train the model.
- Utilise TensorFlow tools to evaluate the trained model and visualise results on TensorBoard.

The datasets used in this implementation include the COCO Dataset (cocodataset.org), which is tightly coupled with the TensorFlow Object Detection API, Google's OpenImages Dataset (https://github.com/openimages/dataset) and the WIDERFACE (http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/) dataset.

As with the other aspects of the system, the algorithm is exposed via a REST endpoint in the OpenAPI standard and simply appears as another service. As access to the procedures is mediated by the REST service, it is also controlled by the security model described in WP3.

2.6 Facial recognition model implementation

A deployed version of the facial recognition model (.pb file) was stored on a Flask (http://flask.pocoo.org/) server within a Docker (https://www.docker.com/) container. The frozen model provides the ability to use the TensorFlow inference functions to identify objects in previously unseen images. Server side functions allow a POST call to send an image to the model which returns the same image annotated with labelled objects, along with annotation information in the response header. This information can then be used edit the image, such as blur or crop out detected faces as required (Figure 7).

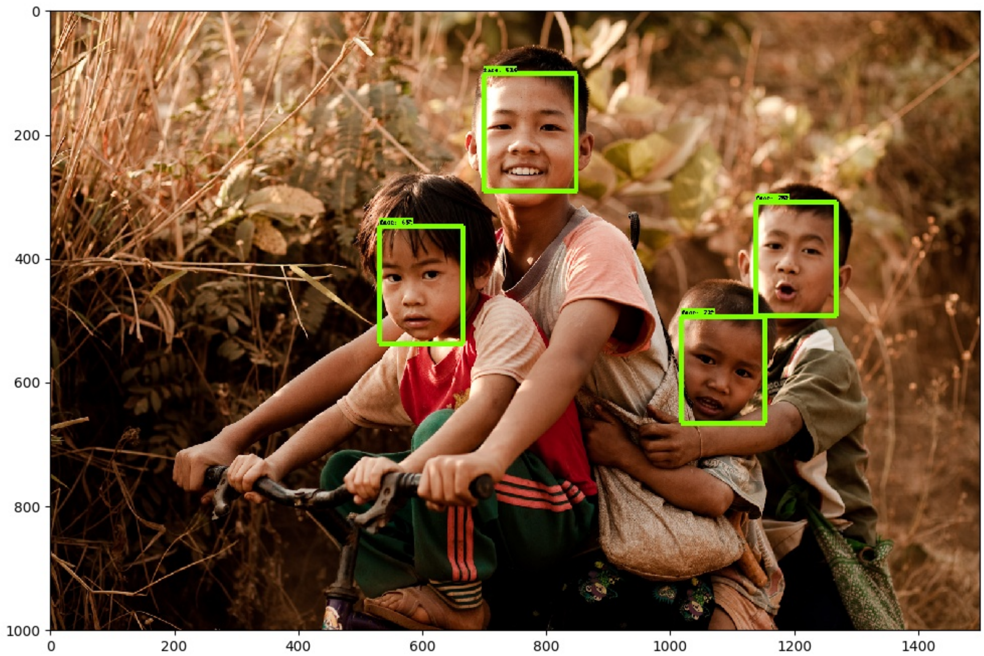

Figure 6 Example output image from the facial recognition service (image take from Open Images Dataset)

The model and label mapping can easily be enhanced with different models or mappings should the use case change or evolve in the future. Additionally, the Flask server has a simple browser implementation to manually test the model and POST operations are working appropriately.

Overall, the QA service for facial recognition is reliable and the concept of doing facial recognition is well-known. A future step is to implement number plate recognition, as this is potentially another identifying mark that contravenes GDPR legislation.

2.6.1 AI for crop recognition

AI for crop recognition differs from the facial recognition use case as it is atypical and models are constructed from training data suited to the use cases. Testing is, therefore, required to understand the practicalities and performance of the developed model for LandSense use cases. This section covers the work done to develop a crop recognition model and test the model against some sample data designed to mimic real submissions.

The TensorFlow framework and the *Inception-v3* model was used as the machine learning engine and model retraining was performed using the crop photographs provided by IIASA. A subset of the provided data was kept aside in an attempt to confirm the accuracy of the image classification using files withheld from the training data (a duplication of the standard behaviour used in model training and completed in the facial recognition section), as well as license free photographs obtained from open sources.

The impact of manipulating features of the photographs to test several theories. Finally, the algorithm was encapsulated within a micro-service container design, providing a mechanism for further research and development of the technique as well as integration with other services using a REST architecture.

2.6.1.1 Model Training

Manual inspection of the supplied crop dataset revealed several issues that we anticipated would reduce the performance of our classification algorithm. These can be summarised as follows:

- Seasonal variations in crop appearance
- Inconsistent angle/framing of crops
- Presence of objects seemingly introduced by photographer
- Low quality photographs (under/over exposed)
- Other items that were inconsistent i.e. fences, buildings, pylons, windmills

The sunflower dataset was the initial focus of model development as flowers are a classical use case for evaluation of machine learning image recognition techniques. This is due to recognisable features such as colours, shape and size. Previous research within the subject area of flower detection provided numerous case studies, techniques and datasets for consideration. One such example can be found in the Open Images (https://storage.googleapis.com/openimages) Dataset produced by Google AI.

Figure 8 demonstrates annotation of sunflowers with object bounding boxes by professional subject matter experts. The photographs vary considerably in several factors; distance and angle from flower, number of flowers, petal direction etc. As such, this data provides an exceptional basis for identification of sunflowers within photographs and could be applied as part of a TensorFlow model.

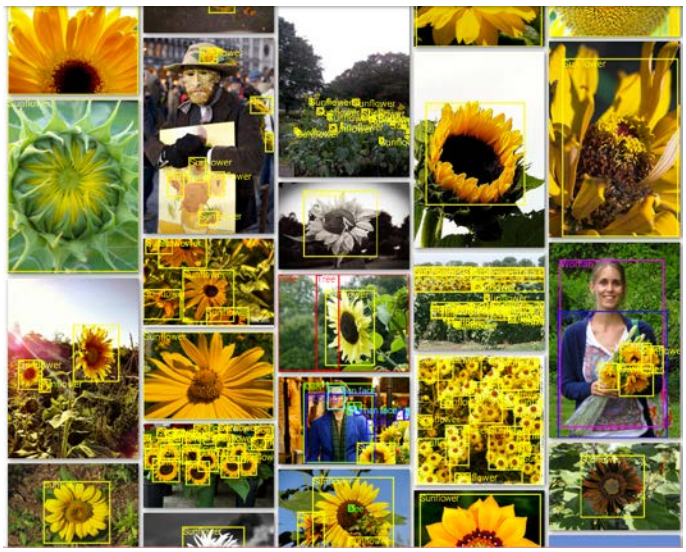

Figure 7 Open Images Dataset sunflower objects

Figure 9 below from the training dataset is immediately recognisable to humans as a field of sunflowers. A TensorFlow based classification algorithm using the Open Images model is able classify this image as containing many sunflowers with a high-degree of confidence.

Figure 8 Yellow sunflower field

Unfortunately, by way of comparison, many of the photographs provided in the training data are vastly different and as such would not be easily identifiable as being sunflower crops by humans, particularly those without knowledge of agriculture. Furthermore, the Open Images model is limited to the actual flower, rather than the entire crop, whereas the majority of sunflower photographs in our training dataset do not show the crops in flower. This is mitigated by sensible selection of data for the training datasets, although in the real world, the provision of the volumes of data required can be challenging, therefore the model trainer is required to work with the data available to produce a model. Figure 10 is particularly problematic, and highly representative of the provided sunflower dataset.

Figure 11 shows a typical issue in that it has mixed content, there is more than one crop type in the image and it is therefore difficult for the model to understand what the focus of the photograph is. Figure 12 shows an obscured photograph, where the camera person has obscured the *useful* parts of the image with another object (a clip board), this maybe a *junk* image or it may have been taken purposely. Regardless, it should be omitted from the model. Figure 13 provides the AI algorithm with useful inputs to build a model, however the photograph differs from others of sunflowers because it was taken prior to flowering. This is common with plants as they have a lifecycle where appearance differs throughout the year. Perhaps a way to mitigate this is to produce several models that are appropriate for different times of year.

Figure 9 Photograph showing sparse filed with no crop above surface

Figure 10 Photograph showing predominantly another crop with sunflowers in the foreground

Figure 11 Photograph showing sunflowers mostly obscured by foreground clipboard

Figure 12 Photograph showing sunflowers prior to flowering

In order to improve the potential accuracy of the sunflower classifier, it was necessary to manually organise the training data into sub-labels of sunflowers to reflect the various stages of the crop's lifecycle. During this process, images of low quality that are atypical of the expected photographs provided for classification were removed.

Other crop types revealed some patterns that enable classification by the human eye (Figures 14 – 19).

Figure 13 Maize showing distance between planting lines

Figure 14 Potato crops with a wide lead coverage and trenches

Figure 15 Rapeseed crops with clearly visible stems

Figure 16 Rapeseed in flower

Figure 17 Wheat shape and colour

Figure 18 Sugar beet with large, overlapping leaves

In general, there are some common traits of the photographs across all of the crop types:

- Landscape orientation
- Sky occupies approximately one-quarter of photograph
- Presence of a (typically red) single object, usually in the middle of the scene. Objects include a clipboard, flag, and cone. These are usually included in a photograph to provide the viewer with a sense of scale of the objects of interest.

These traits do not contribute towards the identification of the crop, therefore it was necessary to determine the extent that these influenced the image classification. The model was initially trained with labels for the crops of sunflower, maize, rapeseed, sugar beet, wheat and potato. After testing this model, a large number of images were collected from freely available sources using the search terms below, before training these as new labels.

- Blue objects
- Brown objects
- Green leaves
- Green objects

- Landscapes
- Soil field
- Sticks
- Trees
- Yellow flowers

After extending our model to include these datasets, the crop labels were selected over these new labels (with one exception, where 'yellow flowers' achieved a higher classification than sunflowers). This suggests that the training dataset is of sufficient size and contains common features that reflect the appearance of crops as opposed to less specific shapes, colours or patterns.

2.6.2 Test Results

The following section provides a summary of the results when tested against images from the IIASA dataset (excluded from the training process), as well as those obtained from open sources. A sample of the images and scores are included and the comments reflect the overall performance against a larger number of photographs than are included in this report. The next subsections show the testing results for each of the classes considered. The results are expressed as a measure of *certainty* where 1 is completely certain and 0 is uncertain. In practice, the software can be configured to accept or reject images based upon a level of certainty. For example, identification of sunflowers is reliable, therefore the acceptance threshold for this plant can be set high whilst maintaining confidence that the system will not produce false negative results (that is, rejecting photographs that match the tagging). Another crop such as potato is not as reliable, therefore the threshold for potato identification would likely have to be lower to reduce the probability of producing false negatives. It maybe that the model is not suitable for use in potato plant identification at all or manual intervention of human users is required to use the system operationally.

2.6.2.1 Sunflowers

The sunflower recognition performed well despite the absence of large yellow flowers within many of the images. In some images of low resolution, the crop was misidentified as rapeseed which does have a similar appearance, particularly from certain distances and angles (Table 2).

Table 2 Results for sunflower recognition

Image	Machine Classification	Human Assessment	Source
	sunflower 0.79 maize 0.11 sugarbeet 0.09	Correct classification	Training data

	sunflower 0.99 potato 0.01	Correct classification	Training data
	sunflower 0.99 potato 0.01	Correct classification	Training data
	sunflower 0.89 potato 0.1	Correct classification	Training data
	sunflower 0.93 potato 0.035 sugarbeet 0.03	Correct classification	Flickr

2.6.2.2 Rapeseed

Rapeseed crops have some distinct features, including the colour of the flowers and the shape of the stem. The classification performed adequately, although incorrectly classified one of the photographs as wheat. Manual inspection revealed that this photograph was of a particularly dry field of rapeseed crops, with no flowers and both the appearance and colour are similar to wheat (Table 3).

This result is illustrative of an error that a non-expert human may repeat and suggests that the model would benefit from having a label for rapeseed with such characteristics .This should lead to a more reliable classification if sufficient images for training can be obtained.

Table 3 Results for rapeseed recognition

Image	Machine Classification	Human Assessment	Source
	wheat 0.55 maize 0.19 rapeseed 0.15 sugarbeet 0.08 sunflower 0.02	Incorrect classification, low confidence	Training data
	rapeseed 0.98 potato 0.01	Correct classification	Training data
	rapeseed 0.98 maize 0.01	Correct classification	Training data
	rapeseed 0.98 wheat 0.01	Correct classification	Training data
	rapeseed 0.48 wheat 0.32 maize 0.1 sugarbeet 0.08 sunflower 0.0	Correct classification	Training data

	rapeseed 0.95 sunflower 0.03	Correct classification	Flickr

2.6.2.3 Maize

In three of the tested images, the algorithm was almost certain that the test images were maize. However, one of the images tested was classified as a potato with a high level of confidence (Table 4). This particular file appeared on inspection do not contain any maize so this may a misleading test case arising from an erroneous photograph. The classifier also failed to confidently classify an image which contained a large amount of soil in the foreground.

Table 4 Results for maize classification

Image	Machine Classification	Human Assessment	Source
	maize 0.59 sugarbeet 0.21 potato 0.14 sunflower 0.03 wheat 0.01	Incorrect classification, low confidence	Training data
	potato 0.92 rapeseed 0.05 sunflower 0.02 maize 0.01	Incorrect classification	Training data
	maize 0.87 sunflower 0.04 sugarbeet 0.03 wheat 0.02 potato 0.02	Correct classification	Training data

	maize 1.0	Correct classification	Training data
	maize 0.99	Correct classification	Training data
	maize 1.0	Correct classification	Flickr

2.6.2.4 Potato

The classification of potatoes proved the most problematic for the first version of our model. A significant improvement in performance was observed when applying a crop to photographs which increased the ratio of correctly classified potatoes from 1/5 to 4/5, with >70% confidence (Table 5).

Table 5 Results for potato classification

Image	Machine Classification	Human Assessment	Source
	wheat 0.37 potato 0.34 rapeseed 0.13 maize 0.1 sunflower 0.03	Unable to classify	Training data

	sugarbeet 0.81 potato 0.13 sunflower 0.03 maize 0.03 rapeseed 0.01	Incorrect classification	Training data
	sugarbeet 0.8 maize 0.11 potato 0.09 sunflower 0.01 wheat 0.0	Incorrect classification	Training data
	potato 0.93 sugarbeet 0.05 maize 0.01 sunflower 0.01	Correct classification	Training data
	potato 0.55 sugarbeet 0.28 rapeseed 0.08 sunflower 0.06 maize 0.03	Correct, low confidence classification	Training data

2.6.2.5 Wheat

The model excelled at wheat crop classification (Table 6), on each occasion correctly classifying images with high-levels of confidence, suggesting the model is deployed successfully without significant tuning of the training data or algorithm required.

Table 6 Results for wheat classification

Image	Machine Classification	Human Assessment	Source
	wheat 0.96 rapeseed 0.03 maize 0.01	Correct classification	Training data
	wheat 1.0	Correct classification	Training data
	wheat 0.98 maize 0.01 sugarbeet 0.01	Correct classification	Training data
	wheat 0.77 rapeseed 0.2 sugarbeet 0.02	Correct classification	Training data
	wheat 0.99 rapeseed 0.01	Correct classification	Training data

	wheat 0.87 rapeseed 0.1 potato 0.01 sugarbeet 0.01	Correct classification	Flickr

2.6.2.6 Sugar beet

The sugar beet classification is problematic for a variety of reasons, however, a number of conclusions can be drawn that allow improvements to be proposed (Table 7).

- The algorithm performed well with mature crops with a photograph taken in fairly close proximity.
- There is some similarity in the appearance of a sugar beet and potato plant, however they have very distinct leaf shapes. It is our expectation that an object detection approach, as outlined in the Open Images sunflower overview, would provide a substantial improvement in this scenario.

Table 7 Results for sugar beet classification

Image	Classification	Performance	Source
	sugarbeet 0.67 maize 0.26 rapeseed 0.03 potato 0.02 sunflower 0.01	Correct classification, low confidence	Training data
	sugarbeet 0.74 potato 0.19 sunflower 0.04 maize 0.02 wheat 0.01	Correct classification	Training data
	potato 0.42 sugarbeet 0.25 wheat 0.09 sunflower 0.09 maize 0.09	Incorrect classification, low confidence	Training data

	sugarbeet 0.75 potato 0.12 rapeseed 0.04 wheat 0.03 sunflower 0.03	Correct classification	Training data
	rapeseed 0.41 maize 0.29 wheat 0.17 potato 0.06 sugarbeet 0.05	Incorrect classification, low confidence	Training data
	sugarbeet 0.85 sunflower 0.08 potato 0.06 maize 0.01	Correct classification	Flickr

3 Photo Manipulation Experimentation

In addition to utilising the provided photography in a model, attempts at modifying the photographs were made to increase the number, variety and utility of the training data. This experiment was carried out in response to the issues identified with the photographs expressed in Section 2.6.1.1 and from the results described in the previous section. This Section discusses some of the experimentation and results completed as part of this endeavour.

3.1 Resizing

An evaluation of the dataset revealed that the majority of the photographs showed the crops occupying approximately 75% of the image, with the sky/horizon accounting for the remainder. Whilst the sky alone should not influence the classification, there are often other patterns that appear in this region of photographs which may impact the model including trees, buildings, pylons, windmills and other structures. The machine learning approach emphasises the common attributes that should be present in the photographs, namely the crops, however there is also wide variation in these as previously outlined.

We used the ImageMagick (https://www.imagemagick.org/script/index.php) convert library to crop of the photographs from the top edge by amounts of 20% and 40% before retraining the model to determine if the performance was impacted. The findings suggest that in many cases, the confidence of classification improved and that images which may have been previously incorrectly classified were more likely to be correctly classified in the cropped dataset. The results were not universally superior, which is likely due to the differences in crop height and photograph framing. For example, with sunflowers, cropping at 40% would occasionally result in the removal of the flowers which are key to the success of the recognition.

In conclusion there is appears to be value in removing irrelevant data from the photographs prior to training and classification, but a more considered approach is required for further improvements to be realised.

3.1.1 Object Insertion

A simple test was performed to establish the influence of the non-crop objects such as discs, clipboards and cones on the classification algorithm.

First, a photograph from the sugar beet test set was classified (Figure 20), with the following scores produced:

Sugar beet 0.75
potato 0.12
rapeseed 0.04

The photograph was then manipulated to contain a disc which was copied from a sunflower photograph (Figure 21), which resulted in the following scores:

Sugar beet 0.45
sunflower 0.24
potato 0.15

Another manipulation was performed using a different disc which had not been obtained from the training data but had similar properties (Figure 22qwf), giving the following scores:

sugarbeet 0.63
rapeseed 0.13
potato 0.12

In the original image, the photograph was classified as a sugar beet, although the confidence rating of 0.75 suggested some uncertainty. In the manipulated image, the score decreased to 0.45, and the probability of the image containing sunflowers increased. In the third image, the sugar beet confidence also decreased, but the impact upon the sunflower classification was negligible.

Additional images were tested that were not from the crop dataset, to ensure the model would not label a photograph containing similar object to those found in the crops as a crop image. This was proven to not be the case. However, testing illustrates that the presence of the objects in the dataset is influencing classification to some extent, and further work is recommend to understand the purpose of the objects and whether detection in order to remove/ignore these features of the photograph is worthwhile.

Figure 19 Original sugar beet photograph

Figure 20 Manipulated sugar beet photograph

Figure 21 Second manipulated photograph

3.1.2 Colour Distribution Analysis

A useful technique in machine processing of images is to examine the dominant colour, as well as the range of colour values (using a colour histogram or other means of representation).

The 'convert' program within the ImageMagick library was used to reduce photographs to a single pixel, before outputting the Red, Green and Blue (RGB) decimal values which range from 0 to 255.

For crops such as sugar beet, maize and potato, the expected dominant colour is green. By examining the RGB values of the photographs, files can be identified that may benefit from manual investigation.

For example, the maize dataset had average values of: Red: 125, Green 136, Blue 107. The following images were amongst those with the highest standard deviation (Figure 23, Figure 24). Values approaching 0,0,0 are closer to black, and values approaching 255,255,255 are closer to white.

Figure 22 Overexposed photograph (Red 183, Green 192, Blue 183)

Figure 23 Underexposed photograph (Red: 30, Green: 32, Blue: 30)

This simple approach could be applied, or extended with more advanced colour histogram analysis, to both improve the quality of the training data set and infer information about photographs that require classification. For example, where the dominant colour of the photograph is brown, this suggests that the crops are immature, that soil is the main feature of the photograph and that the classification will be challenging.

3.2 Recommendations for the crop model

The training data, whilst of a reasonable quantity, lacked quality which reduced the effectiveness of the approach. Furthermore, the variable nature of crops provides a challenging environment for classification of photographs, both by humans and using AI. Despite these limitations, the approach demonstrated that the technique is successful in many circumstances, supporting further research into optimisation and improvement of the model.

It is recommended that the training data is manually reviewed further to remove photographs that do not primarily capture the appearance of the respective crops. Additionally, for the classifier to handle seasonal variations, the photographs should be arranged accordingly into sub-labels. Consistent framing of the photographs should also be sought, including the removal of non-crop objects if possible, otherwise these may require removal via a separate machine learning approach.

The current model expects the photograph uploaded to contain one of six crop varieties and returns scores associated with these labels above a minimum threshold. In testing, we observed that uploads containing unrelated content (for example, household objects) may result in a score similar to that of a genuine crop photograph where there is some doubt over the crop in question. This is not considered to be a limitation based on the proposed application for this technology, however, for other use cases it may be necessary to train the model with alternative datasets and a label of *other*. The requirement for such *negative* training should be reduced by improving the crop sample dataset and increasing the minimum confidence threshold depending on the level of confidence required.

Once the aforementioned measures have been completed, the model should be retrained using the sample scripts provided using a photograph crop of 20% from the top edge. This will likely result in a performant machine learning algorithm which satisfies the aim of classifying user submitted photographs with a high-level of confidence.

4 Conclusion

Overall the LandSense QA services have taken the work done in COBWEB and re-implemented the data quality assessment algorithms taken from ISO 19157 into a standard REST api. The additional requirements from the use cases (referenced in D5.4) have focused qualification of submitted and reported image content and conforming to GDPR legislation. These use cases have been largely approached through the use of AI technologies to aid in image content classification and recognition. The AI work described in this document outlines the challenges faced with recognising the content of imagery when they are crowdsourced and user submitted, however there are methods to mitigate this through training the AI model.

The initial test results documented in this report are promising in the provided use cases. The potential for AI in data qualification goes beyond the simple use cases provided as models can be built to identify and classify content in many different types of data, structured or unstructured. It is the ambition of LandSense to utilise data quality techniques in a multitude of use cases with the ambition of producing data quality *stamps* for supporting metadata documents that will give assurance to users of the data.

5 References

Meek, Sam, Mike J. Jackson, and Didier G. Leibovici. "A flexible framework for assessing the quality of crowdsourced data." (2014).

Meek, Sam, Mike Jackson, and Didier G. Leibovici. "A BPMN solution for chaining OGC services to quality assure location-based crowdsourced data." Computers & Geosciences 87 (2016): 76-83.

Schumann, Guy. Kettner, Albert. Gallagher, J. Potter, N. Meek, S. Hu, L. and Hintz, D. "OGC Testbed-13: NA001 Climate Data Accessibility for Adaptation Planning." OGC Testbed-13 (2018).

[i] https://www.openapis.org/