User manual Nature Identification[1] API

Document management

Name	Date	Version	Author
User manual Nature Identification API	12-11-2018	1.0.	Naturalis Biodiversity Center, Leiden

Table of contents

[1] Identification definition: https://en.wikipedia.org/wiki/Identification_(biology)

Getting started

Nature Identification API

The Nature Identification API offers AI based webservices capable of identifying nature objects on images. Identification is done by species recognition models. Created with Deep learning technology, these models can identify plants and animals on specific taxonomic levels with optimised level of accuracy.

The API currently offers access to an extensive species recognition model for the Netherlands and Belgium based on images of Waarneming.nl and Waarnemingen.be and has been trained with large sets of observation data. The tag for this model is benlall.

Base url

The base url of the API is:

```
https://identify.biodiversityanalysis.nl
```

Terminology

- Identification *model implementations* are published instead of models (a model implementation is a combination of one or more models and a configuration of how to use them)
- <tag> refers to a manually assigned name for a specific model implementation

Species recognition model overview

The API currently offers access to an extensive species recognition model for the Netherlands and Belgium based on images of Waarneming.nl and Waarnemingen.be and has been trained with large sets of observation data.
The tag for this model is benlall.

Webservices overview

Webservice	**Endpoints per service [base url..]**	**Description**
Identify	/v1/observation/identify	Common endpoint for all model implementations
	/v1/observation/identify/<tag>	Endpoint for specific implementation
	/v1/observation/identify/<tag>/auth	Endpoint for specific implementation with user authorization
Taxa	/v1/observation/taxa/<tag>	Endpoint for taxa in implementation
	/v1/observation/taxa/<tag>?id=<id>	Endpoint for taxa in implementation filtered by taxon id

Documentation	/v1/observation/documentation/<tag>	Endpoint for documentation of implementation (description, release notes, etc.)
Endpoints	/v1/observation/endpoints/<tag>	Endpoint that lists all the endpoints for a specific tag
	/v1/observation/endpoints	Endpoint that lists all the endpoints

The v1 API is the first stable version of the API. Updates to the v1 API will be backwards compatible with existing v1 applications.

For the most recent list of available webservices, see list of endpoints (json)

Endpoint service

<table>
<tr><td>title:</td><td><b>Observation Identification Service - Endpoints</b></td></tr>
<tr><td>description:</td><td>provides a list of endpoints</td></tr>
<tr><td>endpoint:</td><td>/v1/observation/endpoints
/v1/observation/endpoints/<tag></td></tr>
<tr><td>method:</td><td>GET</td></tr>
<tr><td>parameters:</td><td>None</td></tr>
<tr><td>response (successful):</td><td>
<p>HTTP code: 200</p>
<p>JSON-structure, containing the following elements:</p>
<ul>
<li><b>endpoints</b>
<ul>
<li><b>documentation</b><ul><li><b>url</b></li></ul></li>
<li><b>endpoints</b><ul><li><b>url</b></li></ul></li>
<li><b>identify</b><ul><li><b>url</b></li></ul></li>
<li><b>identify_base</b><ul><li><b>url</b></li></ul></li>
<li><b>taxa</b><ul><li><b>url</b></li></ul></li>
<li><b>taxa_with_filter</b><ul><li><b>url</b></li></ul></li>
</ul>
</li>
</ul>
<table>
<tr><th><i>JSON field</i></th><th><i>data-type</i></th><th><i>values</i></th><th><i>description</i></th></tr>
<tr><td><b>endpoints.documentation.url</b></td><td>string</td><td>valid URL</td><td>points to the endpoint for documentation</td></tr>
<tr><td><b>endpoints.endpoints.url</b></td><td>string</td><td>valid URL</td><td>points to the endpoint for the endpoints</td></tr>
<tr><td><b>endpoints.identify.url</b></td><td>string</td><td>valid URL</td><td>points to the endpoint for a specific species identification implementation</td></tr>
<tr><td><b>endpoints.identify_base.url</b></td><td>string</td><td>valid URL</td><td>points to the general endpoint for species identification</td></tr>
<tr><td><b>endpoints.taxa.url</b></td><td>string</td><td>valid URL</td><td>points to the endpoint for</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td>
<table>
<tr><td></td><td></td><td></td><td>retrieving the taxa</td></tr>
<tr><td><b>endpoints.taxa_with_filter.url</b></td><td>string</td><td>valid URL</td><td>points to the endpoint for retrieving the taxa based on a filter</td></tr>
</table>
</td>
</tr>
<tr>
<td>response (error):</td>
<td>HTTP code: 400
{
  "error": {
    "code": "model_implementation_not_found",
    "message": "The model implementation with the requested tag does not exist"
  }
}</td>
</tr>
<tr>
<td>example request:</td>
<td>curl https://identify.biodiversityanalysis.nl/v1/observation/endpoints/benlall</td>
</tr>
<tr>
<td>example response:</td>
<td>{
  "endpoints": {
    "documentation": {
      "url": "/v1/observation/documentation/{tag}"
    },
    "endpoints": {
      "url": "/v1/observation/endpoints/{tag}"
    },
    "identify": {
      "url": "/v1/observation/identify/{tag}"
    },
    "identify_base": {
      "url": "/v1/observation/identify"
    },
    "taxa": {
      "url": "/v1/observation/taxa/{tag}"
    },
    "taxa_with_filter": {
      "url": "/v1/observation/taxa/{tag}?id={taxon_id}"
    }
  }
}</td>
</tr>
</table>

Identify webservice

<table>
<tr><td>title:</td><td><b>Observation Identification Service - Identify</b></td></tr>
<tr><td>description:</td><td>provides species identification implementations based on images</td></tr>
<tr><td>endpoints:</td><td>/v1/observation/identify
/v1/observation/identify/<tag></td></tr>
<tr><td>method:</td><td>POST</td></tr>
<tr><td>parameters:</td><td>POST
<b>image</b>: binary image data

URL (path)
<b>tag</b>: [optional] name of implementation to be used, if no name is specified the most appropriate model for the location of the observation will be used. Currently the only implementation is the Dutch/Belgium identification implementation with tag “benlall” so that model will always be used.</td></tr>
<tr><td>response (successful):</td><td>HTTP code: 200

JSON-structure, containing the following elements:
<ul>
<li><b>identification</b><ul><li><b>image</b><ul><li><b>confidence</b></li></ul></li></ul></li>
<li><b>predictions[...]</b><ul><li><b>probability</b></li><li><b>taxon</b><ul><li><b>id</b></li><li><b>name</b></li></ul></li></ul></li>
<li><b>model_implementation</b><ul><li><b>version</b></li></ul></li>
<li><b>links</b><ul><li><b>taxa</b><ul><li><b>url</b></li></ul></li><li><b>taxa_with_filter</b><ul><li><b>url</b></li></ul></li></ul></li>
</ul>
<table>
<tr><th><i>JSON field</i></th><th><i>data-type</i></th><th><i>values</i></th><th><i>description</i></th></tr>
<tr><td><b>identification.image.confidence</b></td><td>string</td><td>“confident” (most likely prediction has a probability > 0.9)
“uncertain” (most likely prediction has a probability > 0.2).
“unknown” (otherwise)</td><td>indicates verbally the confidence of the image based species identification</td></tr>
<tr><td><b>predictions</b></td><td>array</td><td>top 10 predictions, ordered on probability, highest first.</td><td>most likely predictions</td></tr>
<tr><td><b>predictions[...].probability</b></td><td>float</td><td>[0.0, 1.0]</td><td>probability of match for taxon</td></tr>
</table>
</td></tr>
</table>

predictions[...].taxon.id	string	unique taxon identifier in format {id_at_source}@{source_identifier}	identifies taxa in and across sources **NB** source_identifier = “WRN” for waarneming.nl
predictions[...].taxon.name	string	scientific name of the taxon	accepted scientific name for taxon
model_implementation.version	string	unique model implementation identifier	see values
links.taxa.url	string	valid URL	points to the endpoint for retrieving the taxa
links.taxa_with_filter.url	string	valid URL	points to the endpoint for retrieving the taxa based on a filter

response (error):

```
HTTP code: 400
{
  "error": {
    "code": "model_implementation_not_found",
    "message": "The model implementation with the requested tag does not exist"
  }
}

HTTP code: 400
{
  "error": {identifications _
    "code": "received_no_files",
    "message": "Did not receive any files"
  }
}

HTTP code: 405
{
  "error": {
    "code": "method_not_allowed",
    "message": "Method not allowed (only supports POST)"
  }
}

HTTP code: 415
{
  "error": {
    "code": "unsupported_media_type",
    "message": "Unsupported media type"
  }
}

HTTP code: 500
{
  "error": {
    "code": "general_server_error",
    "message": "General server error"
```

	} }
example request:	curl -X POST -F "image=@sperwer_1.jpg" -F "image=@sperwer_2.jpg" https://identify.biodiversityanalysis.nl/v1/observation/identify

example response:	{
	  "identification": {
	    "image": {
	      "confidence": "confident"
	    }
	   },
	  "predictions": [
	    {
	      "probability": 0.9970064163208008,
	      "taxon": {
	        "id": "178@WRN",
	        "name": "Accipiter nisus"
	      }
	    },
	    {
	      "probability": 0.0029197214171290398,
	      "taxon": {
	        "id": "117@WRN",
	        "name": "Accipiter gentilis"
	      }
	    },
	    "...",
	    {
	      "probability": 2.6637922090344546e-08,
	      "taxon": {
	        "id": "326@WRN",
	        "name": "Falco columbarius"
	      }
	    }
	  ],
	  "links": {
	    "taxa": {
	      "url": "/v1/observation/taxa/benlall"
	    },
	    "taxa_with_filter": {
	      "url": "/v1/observation/taxa/benlall?id={taxon_id}"
	    }
	  },
	  "model_implementation": {
	    "version": "1fd68f8c8cb93ec4e45049fcf9a056628e9599aa815790a2a7b568aa"
	  }
	}

Taxa webservice

<table>
<tr><td>title:</td><td><b>Observation Identification Service - Taxa</b></td></tr>
<tr><td>description:</td><td>lists the taxa that are included in an identification implementation</td></tr>
<tr><td>endpoint:</td><td>/v1/observation/taxa/<tag></td></tr>
<tr><td>method:</td><td>GET</td></tr>
<tr><td>parameters:</td><td>URL (query)
<b>id:</b> [optional] query field for taxon id, filters the output list</td></tr>
<tr><td>response (successful):</td><td>HTTP code: 200

JSON-structure, containing the following elements:
<ul>
<li><b>taxa[...]</b>
<ul>
<li><b>id</b></li>
<li><b>name</b></li>
<li><b>vernacular_names</b></li>
<li><b>alternate_ids</b></li>
</ul></li>
<li><b>model_implementation</b>
<ul>
<li><b>version</b></li>
</ul></li>
</ul>
<table>
<tr><th><i>JSON field</i></th><th><i>data-type</i></th><th><i>values</i></th><th><i>description</i></th></tr>
<tr><td><b>taxa</b></td><td>array</td><td>list of records of taxa included in implementation</td><td>see values</td></tr>
<tr><td><b>taxa[...].id</b></td><td>string</td><td>unique taxon identifier in format {id_at_source}@{source_identifier}</td><td>identifies taxa in and across sources
<b>NB</b> source_identifier = “WRN” for waarneming.nl</td></tr>
<tr><td><b>taxa[...].name</b></td><td>string</td><td>scientific name of the taxon</td><td>accepted scientific name for taxon</td></tr>
<tr><td><b>taxa[...]. alternate_ids</b></td><td>list</td><td>list of qualified taxon ids, see <b>.id</b></td><td>IDs of the taxon in other sources such as soortenregister.nl, can be empty</td></tr>
<tr><td><b>taxa[...]. vernacular_names</b></td><td>map</td><td><i>key</i>: language code
<i>value</i>: vernacular name</td><td>vernacular names for taxon, can be empty</td></tr>
<tr><td><b>model_implementation. version</b></td><td>string</td><td>unique implementation identifier</td><td>see values</td></tr>
</table></td></tr>
<tr><td>response (error):</td><td>HTTP code: 400
{
  "error": {</td></tr>
</table>

	"code": "model_implementation_not_found", "message": "The model implementation with the requested tag does not exist" } }
example request:	curl https://identify.biodiversityanalysis.nl/v1/observation/taxa/benlall curl https://identify.biodiversityanalysis.nl/v1/observation/taxa/benlall?id=20796@WRN

example response:	{
	"taxa": [
	{
	"id": "20796@WRN",
	"name": "Abax parallelepipedus",
	"vernacular_names": {
	"nl": "Bosbulldozer"
	},
	alternate_ids: []
	},
	{
	"id": "153533@WRN",
	"name": "Abia aenea",
	"vernacular_names": {},
	alternate_ids: []
	}
	],
	"model_implementation": {
	"version": "1fd68f8c8cb93ec4e45049fcf9a056628e9599aa815790a2a7b568aa"
	}
	}

Documentation webservice

title:	**Observation Identification Service - Documentation**
description:	provides structured documentation for model
endpoint:	/v1/observation/documentation/<tag>
method:	GET
parameters:	None
response (successful):	HTTP code: 200 JSON-structure, containing the following elements: ● **release_notes** ● **description** ● **use_policy** ○ **unauthorized** ■ **max_per_day** ○ **authorized** ■ **max_per_day** ■ **url** ■ **authentication** ● **model_implementation:** ○ **class_count** ○ **version**

JSON field	***data-type***	***values***	***description***
release_notes	map	*key*: language code *value*: JSON escaped HTML	Release notes indicate changes a.o. changes w.r.t. to previous models
description	map	*key*: language code *value*: JSON escaped HTML	The description is a natural language description of what the model caurl https://identify.biodiversityanalysis.nl/v1/observation/documentation/benlalln do (and can't)
use_policy.unauthorized.max_per_day	int	[0, 2^32 - 1]	Maximum number of identifications per day without authentication
use_policy.authorized.max_per_day	string	"fair use"	Indicates the conditions of use of the API when authorized
use_policy.authorized.url	string	valid URL	URL to the

			authorized version of the service
use_policy.authorized. authentication	string	"https://tools.ietf.org/html/rfc7617"	describes the type of authentication used
model_implementation. version	string	unique implementation identifier	see values
model_implementation. class_count	int	[2, 2^32 - 1]	Number of classes (taxa) in implementation

response (error):	HTTP code: 400
	{
	"error": {
	"code": "model_implementation_not_found",
	"message": "The model implementation with the requested tag does not exist"
	}
	}
	url https://identify.biodiversityanalysis.nl/v1/observation/documentation/benlall

example request:	curl https://identify.biodiversityanalysis.nl/v1/observation/documentation/benlall

example response:	{
	"description": {
	"en": "The webservice identifies species of plants, animals and fungi that naturally occur in The Netherlands and Belgium,\n based on photos of observations. The underlying recognition model has been trained using photos from Waarneming.nl. The model contains\n 16148 taxa: 13767 species, 1530 species-groups, 734 subspecies and 117 hybrid species. Identifications can benefit from multiple photos\n from the same observation. Provide up to four photos in a single request to improve identification results."
	},
	"model_implementation": {
	"class_count": 16148,
	"version": "0ff1954435b340fbef20e27e9d892d95f38f53754d576c529b6e54ac"
	},
	"release_notes": {
	"en": "release candidate for public release"
	},
	"use_policy": {
	"authorized": {
	"authentication": "https://tools.ietf.org/html/rfc7617",
	"max_per_day": "fair use",
	"url": "https://identify.biodiversityanalysis.nl/v1/observation/identify/benlall/auth"
	},
	"unauthorized": {
	"max_per_day": 10

	`    }`
	`  }`
	`}`

Contact and support

For information on the species recognition model of waarneming.nl and webAPI access to this model please contact: info@waarneming.nl

For technical information and support on the API please contact: support@naturalis.nl