

WAGRI病虫害AI画像診断 API

2026/3/31

Table of contents

1. 概要	3
1.1 WAGRI 病虫害 AI 画像診断 API	3
2. APIの利用方法	4
2.1 API の利用方法	4
3. リクエストヘッダー/ボディ	7
3.1 共通リクエストヘッダー	7
3.2 共通リクエストボディ	8
4. 現行(2026/3/3リリース)版判別器	13
4.1 分類検出型病虫害判別器	13
5. 現行(2022/04/28リリース)版判別器	40
5.1 分類型病虫害判別器	40
5.2 人によるアノテーション情報を登録する場合	51
6. エラー発生時のレスポンス	55
6.1 エラー発生時のレスポンス	55

1. 概要

1.1 WAGRI 病虫害 AI 画像診断 API

WAGRI 病虫害 AI 画像診断 API は、農研機構が農林水産省戦略的プロジェクト研究推進事業「AI を活用した病虫害診断技術の開発」および内閣府官民研究開発投資拡大プログラム(PRISM)を通じて開発を行った病虫害判別器を利用するための API です。現在、重要園芸、畑作、果樹12作目（トマト・イチゴ・キュウリ・ナス・モモ・ブドウ・タマネギ・ピーマン・カボチャ・エダマメ・ジャガイモ・キク）の主要な病虫害判定を提供しています。

本 API で提供している判別器は、継続的に精度向上のための改良を行います。加えて、判別可能な病虫害のカテゴリについても順次追加を行う予定です。

1.1.1 利用条件

本 API をご利用いただくには、以下の条件を全て満たしている必要があります。

- WAGRI 利用会員であること
- AI 病虫害画像診断 API の利用申請を行い、承認を得ていること

1.1.2 利用申請手続

本 API を利用するためには、本 API の申込時に提示いたします AI 病虫害画像診断 API 利用規約に同意いただき、WAGRI 病虫害 AI 画像診断 API 同意書兼利用申請書を提出いただく必要があります。

利用申請手続の詳細や WAGRI 病虫害 AI 画像診断 API 利用規約などについては、[連絡先](#)にお問い合わせ下さい。

1.1.3 連絡先

農研機構 農業情報研究センター 農業データ連携基盤研究室

sh-wagri@naro.go.jp

2. APIの利用方法

2.1 API の利用方法

画像データを含む JSON を POST することで、病虫害画像診断を行うことが可能です。

リクエストの仕方やレスポンスの構造については、「リクエストヘッダー/ボディ」、「現行版判別器」を参照して下さい。

2.1.1 利用可能な判別器

現行(2026/03/03 リリース)版判別器

現行(2026/03/03 リリース)版判別器 API の URL は以下に示す形式になります。

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/<engine>
```

例えば、分類検出型病中害判別器[キュウリ・葉表裏](POST `naro/rcait-v2/cucumber/leaf`)を使用したい場合、

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/cucumber/leaf
```

となります。

<engine> のパスを変更することで、作目、部位、判別可能カテゴリ、判別器の型を変更することが可能です。

現在公開されている <engine> の一覧を以下に示します。

<engine>	提供元	作目	部位	判別可能カテゴリ	判別器の型
<code>naro/rcait-v2/chrysanthemum/leaf</code>	農研機構	キク	葉表裏	病虫害	分類検出型
<code>naro/rcait-v2/cucumber/leaf</code>	農研機構	キュウリ	葉表裏	病虫害	分類検出型
<code>naro/rcait-v2/cucumber/fruit</code>	農研機構	キュウリ	果実	病虫害	分類検出型
<code>naro/rcait-v2/cucumber/flower</code>	農研機構	キュウリ	花	病虫害	分類検出型
<code>naro/rcait-v2/eggplant/leaf</code>	農研機構	ナス	葉表裏	病虫害	分類検出型
<code>naro/rcait-v2/eggplant/fruit</code>	農研機構	ナス	果実	病虫害	分類検出型
<code>naro/rcait-v2/eggplant/flower</code>	農研機構	ナス	花	病虫害	分類検出型
<code>naro/rcait-v2/grape/leaf</code>	農研機構	ブドウ	葉表裏	病虫害	分類検出型
<code>naro/rcait-v2/grape/fruit</code>	農研機構	ブドウ	果実	病虫害	分類検出型
<code>naro/rcait-v2/greenpepper/leaf</code>	農研機構	ピーマン	葉表裏	病虫害	分類検出型
<code>naro/rcait-v2/onion/leaf</code>	農研機構	タマネギ	葉表裏	病虫害	分類検出型
<code>naro/rcait-v2/peach/leaf</code>	農研機構	モモ	葉表裏	病虫害	分類検出型

<engine>	提供元	作目	部位	判別可能カテゴリ	判別器の型
naro/rcait-v2/peach/fruit	農研機構	モモ	果実	病虫害	分類検出型
naro/rcait-v2/potato/leaf	農研機構	ジャガイモ	葉表裏	病虫害	分類検出型
naro/rcait-v2/pumpkin/leaf	農研機構	カボチャ	葉表裏	病虫害	分類検出型
naro/rcait-v2/soybean/leaf	農研機構	エダマメ	葉表裏	病虫害	分類検出型
naro/rcait-v2/strawberry/leaf	農研機構	イチゴ	葉表裏	病虫害	分類検出型
naro/rcait-v2/strawberry/fruit	農研機構	イチゴ	果実	病虫害	分類検出型
naro/rcait-v2/strawberry/flower	農研機構	イチゴ	花	病虫害	分類検出型
naro/rcait-v2/tomato/leaf	農研機構	トマト	葉表裏	病虫害	分類検出型
naro/rcait-v2/tomato/fruit	農研機構	トマト	果実	病虫害	分類検出型
naro/rcait-v2/tomato/flower	農研機構	トマト	花	病虫害	分類検出型

現行(2022/04/28 リリース)版判別器

現行(2022/04/28 リリース)版判別器 API の URL は以下に示す形式になります。

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/<engine>
```

例えば、分類型病害判別器[キク・部位指定なし](POST naro/rcait-v1/chrysanthemum)を使用したい場合、

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/chrysanthemum
```

となります。

<engine> のパスを変更することで、作目、部位、判別可能カテゴリ、判別器の型を変更することが可能です。

現在公開されている <engine> の一覧を以下に示します。

<engine>	提供元	作目	部位	判別可能カテゴリ	判別器の型
naro/rcait-v1/chrysanthemum	農研機構	キク	部位指定なし	病虫害	分類型
naro/rcait-v1/grape	農研機構	ブドウ	部位指定なし	病虫害	分類型
naro/rcait-v1/greenpepper	農研機構	ピーマン	部位指定なし	病虫害	分類型
naro/rcait-v1/onion	農研機構	タマネギ	部位指定なし	病虫害	分類型
naro/rcait-v1/peach	農研機構	モモ	部位指定なし	病虫害	分類型
naro/rcait-v1/potato	農研機構	ジャガイモ	部位指定なし	病虫害	分類型
naro/rcait-v1/pumpkin	農研機構	カボチャ	部位指定なし	病虫害	分類型

<engine>	提供元	作目	部位	判別可能カテゴリ	判別器の型
naro/rcait-v1/ soybean	農研機構	ダイズ	部位指定なし	病虫害	分類型

3. リクエストヘッダー/ボディ

3.1 共通リクエストヘッダー

どの判別器を使用する場合でも、リクエストヘッダーに設定する必要がある項目は同じです。

Data Type は、[ECMA-404 2nd Edition](#)を参考に定義しています。

Parameter	Data Type	Required	Description
X-User-Id	string	required	エンドユーザーの ID を指定します。 エンドユーザーとは、本 API を契約していただいたベンダー様が管理するユーザーのことです。 値は任意ですが、なるべく個別の ID を設定してください。
Authorization	string	required	API利用者登録時にお知らせしたドメイン名を設定してください。
X-Authorization	string	required	WAGRI 認証用のトークンを指定します。 本トークンの取得方法は、 WAGRI API 利用マニュアル 3.1.2 「アクセストークンの取得方法」 を参照してください。
Content-Type	string	required	リクエストのボディとして送信するデータの種類や形式を指定します。 本 API では、 <code>application/json</code> のみ対応しています。

3.2 共通リクエストボディ

どの判別器を使用する場合でも、リクエストボディに設定可能な項目は同じです。

Data Typeは、[ECMA-404 2nd Edition](#)を参考に定義しています。

リクエストサンプル

Request JSON: AIによる病虫害画像診断を行う場合

```
{
  "assets": [
    {
      "id": "<This parameter is not required.>",
      "X-User-Id": "<This parameter is not required.>",
      "comments": "<This parameter is not required.>",
      "options": {
        "diagnosis": true,
        "attach_image": false,
        "response_json_format": "2022-04-28_classifier_style"
      },
      "images": [
        {
          "filename": "<This parameter is not required.>",
          "filedate": "<This parameter is not required.>",
          "data": "/9j/4AAQSkZJRgA... (base64 encoded image data)"
        }
      ]
    }
  ]
}
```

Request JSON: 人によるアノテーション情報を登録する場合

```
{
  "assets": [
    {
      "id": "<This parameter is not required.>",
      "X-User-Id": "<This parameter is not required.>",
      "comments": "<This parameter is not required.>",
      "options": {
        "diagnosis": false,
        "attach_image": false
      },
      "images": [
        {
          "filename": "<This parameter is not required.>",
          "filedate": "<This parameter is not required.>",
          "data": "/9j/4AAQSkZJRgA... (base64 encoded image data)",
          "annotations": [
            {
              "type": "human_classification",
              "annotator": "<annotator name>",
              "label": [
                {
                  "plant": "cucumber",
                  "class": "斑点細菌病",
                  "score": 9999,
                  "place": "36.02790370512755, 140.10198705396417",
                  "boundingBoxes": [
                    {
                      "bbox": [
                        660,
                        685,
                        721,
                        771
                      ],
                      "byhand": true,
                      "comment": "<Comment>"
                    }
                  ]
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

3.2.1 body (リクエストボディ)

Parameter	Data Type	Required	Description
assets	array	required	リクエスト情報を指定します。 詳細は、 <code>body["assets"][0]</code> をご覧ください。

3.2.2 body["assets"][0]

Parameter	Data Type	Required	Description
id	string	optional	リクエストしたデータのIDを指定します。 本APIを契約していただいたベンダー様が自由に指定してください。
X-User-Id	string	optional	エンドユーザーのIDを指定します。 エンドユーザーとは、本APIを契約していただいたベンダー様が管理するユーザーのことです。 指定する場合はリクエストヘッダーで指定した X-User-Id と同じ値にすることをお勧めします。
comments	string	optional	リクエストへのコメントを指定します。 本APIを契約していただいたベンダー様が自由に指定してください。
options	object	optional	リクエスト時の追加設定を指定します。 詳細は、 <code>body["assets"][0]["options"]</code> を参照してください。
images	array	required	画像情報を指定します。 詳細は、 <code>body["assets"][0]["images"][0]</code> を参照してください。

3.2.3 body["assets"][0]["options"]

Parameter	Data Type	Required	Description
diagnosis	boolean	optional	AIによる病虫害画像診断を行うかどうかについて、指定します。 <code>true</code> のときAIによる病虫害画像診断を行い、 <code>false</code> のときAIによる病虫害画像診断を行いません。 <code>false</code> を指定すると、人によるアノテーション情報を本APIにアップロードすることが可能です。 何も指定しない場合、本APIでは <code>true</code> と判断され、AIによる病虫害画像診断を行います。
attach_image	boolean	optional	レスポンス情報に、リクエスト時の画像を添付するかどうかについて、指定します。 <code>true</code> のときレスポンス情報にリクエスト時の画像を添付し、 <code>false</code> のときレスポンス情報にリクエスト時の画像を添付しません。 何も指定しない場合、本APIでは <code>false</code> と判断され、レスポンス情報にリクエスト時の画像を添付しません。 NOTE: 本オプションは、現行(2022/04/28リリース)版判別器のみ有効です。旧版判別器では指定しても反映されません。

Parameter	Data Type	Required	Description
response_json_format	string	optional	<p>判定結果のレスポンスJSONのフォーマットを指定することが出来ます。</p> <p>2022-04-28_classifier_style を指定すると、旧版判別器のレスポンスJSONを現行(2022/04/28リリース)版分類型病害判別器と同じフォーマットに変換して結果を返します。</p> <p>original_style を指定すると、旧版判別器のフォーマットで返します。</p> <p>何も指定しない場合、本APIではoriginal_style と判断され、旧版判別器のフォーマットで返します。</p> <p>NOTE: 本オプションは旧版判別器のみ有効です。 現行(2022/04/28リリース)版判別器では指定しても反映されません。</p>

3.2.4 body["assets"][0]["images"][0]

Parameter	Data Type	Required	Description
filename	string	optional	画像のファイル名を指定します。
filedate	string	optional	画像の撮影日時を指定します。
data	string	required	<p>base64エンコードした画像データを指定します。</p> <p>NOTE: AIによる病虫害画像診断を行う場合は、最低限このパラメータだけ有効な値であれば、判定結果が返ってきます。</p>
annotations	array	optional (body["assets"][0]["options"]["diagnosis"] が false の場合, required)	<p>アノテーション情報を指定します。</p> <p>body["assets"][0]["options"]["diagnosis"] が true の場合 (AIによる病虫害画像診断を行う場合)、指定する必要はありません(指定しても問題ありませんが、無視されます)。</p> <p>body["assets"][0]["options"]["diagnosis"] が false の場合 (人によるアノテーション情報を登録する場合)には必ず指定する必要があります。</p> <p>詳細は、 body["assets"][0]["images"][0]["annotations"][0] を参照してください。</p>

3.2.5 body["assets"][0]["images"][0]["annotations"][0]

Parameter	Data Type	Required	Description
type	string	optional (body["assets"][0]["options"]["diagnosis"] が false の場合, required)	<p>判定者のタイプを指定します。</p> <p>body["assets"][0]["options"]["diagnosis"] が false (AIによる病虫害画像診断を行わない)の場合には human_classification を指定してください。</p>
annotator	string	optional	判定者情報を指定します。
labels	array	optional (body["assets"][0]["options"]["diagnosis"] が false の場合, required)	<p>アノテーションラベル情報を指定します。</p> <p>body["assets"][0]["options"]["diagnosis"] が false (AIによる病虫害画像診断を行わない)の場合には必ず指定する必要があります。</p> <p>詳細は、 body["assets"][0]["images"][0]["annotations"][0]["labels"][i] を参照して下さい。</p>

3.2.6 body["assets"][0]["images"][0]["annotations"][0]["labels"][i]

Parameter	Data Type	Required	Description
plant	string	optional (body["assets"][0]["options"] ["diagnosis"] が false の場合, required)	作目名を指定します。 body["assets"][0]["options"] ["diagnosis"] が false (AIによる 病虫害画像診断を行わない)の場合には必ず指定する 必要があります。
class	string	optional (body["assets"][0]["options"] ["diagnosis"] が false の場合, required)	病虫害名を指定します。 body["assets"][0]["options"] ["diagnosis"] が false (AIによる 病虫害画像診断を行わない)の場合には必ず指定する 必要があります。
score	number	optional (body["assets"][0]["options"] ["diagnosis"] が false の場合, required)	確度を指定します。 body["assets"][0]["options"] ["diagnosis"] が false (AIによる 病虫害画像診断を行わない)の場合、且つ、専門家による アノテーションの場合には 9999 を指定してください。
place	string	optional	撮影場所の情報を指定します。
boundingBoxes	array	optional	バウンディングボックスの情報を指定します。 バウンディングボックスのアノテーションも人によって 行う場合は、こちらの情報を指定してください。 詳細は、 body["assets"][0]["images"][0]["annotations"][0] ["labels"][i]["boundingBoxes"][0] を参照してください。

3.2.7 body["assets"][0]["images"][0]["annotations"][0]["labels"][i]["boundingBoxes"][0]

Parameter	Data Type	Required	Description
bbox	array	optional	バウンディングボックスの座標情報を指定します。 詳細は、 body["assets"][0]["images"][0]["annotations"][0] ["labels"][i]["boundingBoxes"][0]["bbox"]<index> を参照 してください。
byhand	boolean	optional	人によるバウンディングボックスのアノテーション情報 かどうかを指定します。 人によるアノテーションの場合は true を指定し、AIによる アノテーションの場合は false を指定してください。
comment	string	optional	バウンディングボックス情報へのコメントを指定しま す。

3.2.8 body["assets"][0]["images"][0]["annotations"][0]["labels"][i]["boundingBoxes"][0]["bbox"]<index>

リクエスト時に指定した判別器が判定した結果のバウンディングボックスの座標情報を指定します。

画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義してください。

<index>	Data Type	Description
0	number	バウンディングボックスの始点ピクセルのx座標値を指定します。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で 定義してください。
1	number	バウンディングボックスの始点ピクセルのy座標値を指定します。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で 定義してください。
2	number	バウンディングボックスの終点ピクセルのx座標値を指定します。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で 定義してください。

<index>	Data Type	Description
3	number	バウンディングボックスの終点ピクセルのy座標値を指定します。画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義してください。

4. 現行(2026/3/3リリース)版判別器

4.1 分類検出型病虫害判別器

4.1.1 現行(2026/03/03リリース)版分類検出型病虫害判別器

分類検出型病害判別器では、POSTされた画像に写っている病虫害の分類結果と、POSTされた画像上から病害の部位を検出し、画像上のどの位置に存在するかを返します。また、同一URLのGETメソッドを用いることで、判定結果候補となる病虫害のリストを返します。現行(2026/03/03リリース)版分類検出型病虫害判別器は、以下の判別器のことを指します。

- ・分類検出型病虫害判別器[キク・葉表裏] (POST naro/rcait-v2/chrysanthemum/leaf)
- ・分類検出型病虫害判別器[キュウリ・葉表裏] (POST naro/rcait-v2/cucumber/leaf)
- ・分類検出型病虫害判別器[キュウリ・果実] (POST naro/rcait-v2/cucumber/fruit)
- ・分類検出型病虫害判別器[キュウリ・花] (POST naro/rcait-v2/cucumber/flower)
- ・分類検出型病虫害判別器[ナス・葉表裏] (POST naro/rcait-v2/eggplant/leaf)
- ・分類検出型病虫害判別器[ナス・果実] (POST naro/rcait-v2/eggplant/fruit)
- ・分類検出型病虫害判別器[ナス・花] (POST naro/rcait-v2/eggplant/flower)
- ・分類検出型病虫害判別器[ブドウ・葉表裏] (POST naro/rcait-v2/grape/leaf)
- ・分類検出型病虫害判別器[ブドウ・果実] (POST naro/rcait-v2/grape/fruit)
- ・分類検出型病虫害判別器[ピーマン・葉表裏] (POST naro/rcait-v2/peenpepper/leaf)
- ・分類検出型病虫害判別器[タマネギ・葉表裏] (POST naro/rcait-v2/onion/leaf)
- ・分類検出型病虫害判別器[モモ・葉表裏] (POST naro/rcait-v2/peach/leaf)
- ・分類検出型病虫害判別器[モモ・果実] (POST naro/rcait-v2/peach/fruit)
- ・分類検出型病虫害判別器[ジャガイモ・葉表裏] (POST naro/rcait-v2/potato/leaf)
- ・分類検出型病虫害判別器[カボチャ・葉表裏] (POST naro/rcait-v2/pumpkin/leaf)
- ・分類検出型病虫害判別器[エダマメ・葉表裏] (POST naro/rcait-v2/soybean/leaf)
- ・分類検出型病虫害判別器[イチゴ・葉表裏] (POST naro/rcait-v2/strawberry/leaf)
- ・分類検出型病虫害判別器[イチゴ・果実] (POST naro/rcait-v2/strawberry/fruit)
- ・分類検出型病虫害判別器[イチゴ・花] (POST naro/rcait-v2/strawberry/flower)
- ・分類検出型病虫害判別器[トマト・葉表裏] (POST naro/rcait-v2/tomato/leaf)
- ・分類検出型病虫害判別器[トマト・果実] (POST naro/rcait-v2/tomato/fruit)
- ・分類検出型病虫害判別器[トマト・花] (POST naro/rcait-v2/tomato/flower)

リクエスト

ヘッダー

[共通リクエストヘッダー](#)を参照してください。

ボディ

[共通リクエストボディ](#)を参照してください。

レスポンス

ステータスコード

成功時

200

エラー発生時

エラー発生時のレスポンスを参照してください。

ボディ

AIによる病虫害画像診断を行う場合

現行(2026/03/03リリース)版分類検出型病虫害判別器 - レスポンスボディを参照してください。

人によるアノテーション情報を登録する場合

人によるアノテーション情報を登録する場合 - レスポンスボディを参照してください。

サンプルコード

AIによる病虫害画像診断を行う場合

現行(2026/03/03リリース)版分類型病虫害判別器 - サンプルコードをご覧ください。

人によるアノテーション情報を登録する場合

人によるアノテーション情報を登録する場合 - サンプルコードをご覧ください。

分類検出型病虫害判別器[キク・葉表裏] (POST naro/rcait-v2/chrysanthemum/leaf)

農研機構が開発したキクの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/chrysanthemum/leaf
```

判別結果候補

disease

- ・健全
- ・えそ病
- ・半身萎凋病
- ・白さび病
- ・アザミウマ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉 (表)
- ・葉 (裏)

分類検出型病虫害判別器[キュウリ・葉表裏] (POST naro/rcait-v2/cucumber/leaf)

農研機構が開発したキュウリの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/cucumber/leaf
```

判別結果候補

disease

- 健全
- CCYV
- CMV
- MYSV
- WMV
- ZYMV
- うどんこ病
- すず病
- つる枯病
- べと病
- 斑点細菌病
- 灰色かび病
- 炭疽病
- 褐斑病
- アザミウマ類
- アブラムシ類
- ウリキンウワバ
- オオタバコガ
- コナジラミ類
- チャノホコリダニ
- ハスモンヨトウ
- ハダニ類
- ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- 葉（表）
- 葉（裏）

分類検出型病虫害判別器[キュウリ・果実] (POST naru/rcait-v2/cucumber/fruit)

農研機構が開発したキュウリの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/cucumber/fruit
```

判別結果候補

disease

- 健全

- CCYV
- CMV
- MYSV
- WMV
- ZYMV
- うどんこ病
- すず病
- つる枯病
- べと病
- 斑点細菌病
- 灰色かび病
- 炭疽病
- 褐斑病
- アザミウマ類
- アブラムシ類
- ウリキンウワバ
- オオタバコガ
- コナジラミ類
- チャノホコリダニ
- ハスモンヨトウ
- ハダニ類
- ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- 果実

分類検出型病害判別器[キュウリ・花] (POST naro/rcait-v2/cucumber/flower)

農研機構が開発したキュウリの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/cucumber/flower
```

判別結果候補

disease

- 健全
- CCYV
- CMV
- MYSV
- WMV
- ZYMV

- ・うどんこ病
- ・すす病
- ・つる枯病
- ・べと病
- ・斑点細菌病
- ・灰色かび病
- ・炭疽病
- ・褐斑病
- ・アザミウマ類
- ・アブラムシ類
- ・ウリキンウワバ
- ・オオタバコガ
- ・コナジラミ類
- ・チャノホコリダニ
- ・ハスモンヨトウ
- ・ハダニ類
- ・ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・花

分類検出型病害判別器[ナス・葉表裏] (POST `naro/rcait-v2/eggplant/leaf`)

農研機構が開発したナスの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/eggplant/leaf
```

判別結果候補

disease

- ・健全
- ・うどんこ病
- ・すすかび病
- ・すす病
- ・半身萎凋病
- ・灰色かび病
- ・褐色円星病
- ・褐色腐敗病
- ・青枯病
- ・黒枯病
- ・アザミウマ類
- ・アブラムシ類

- ・オオタバコガ
- ・オオニジュウヤホシテントウ
- ・カメムシ類
- ・コガネムシ類
- ・コナジラミ類
- ・チャノホコリダニ
- ・ハスモンヨトウ
- ・ハダニ類
- ・ハモグリバエ類
- ・ヨトウガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病虫害判別器[ナス・果実] (POST naro/rcait-v2/eggplant/fruit)

農研機構が開発したナスの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/eggplant/fruit
```

判別結果候補

disease

- ・健全
- ・うどんこ病
- ・すすかび病
- ・すす病
- ・半身萎凋病
- ・灰色かび病
- ・褐色円星病
- ・褐色腐敗病
- ・青枯病
- ・黒枯病
- ・アザミウマ類
- ・アブラムシ類
- ・オオタバコガ
- ・オオニジュウヤホシテントウ
- ・カメムシ類
- ・コガネムシ類
- ・コナジラミ類
- ・チャノホコリダニ

- ・ハスモンヨトウ
- ・ハダニ類
- ・ハモグリバエ類
- ・ヨトウガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・果実

分類検出型病害判別器[ナス・花] (POST naro/rcait-v2/eggplant/flower)

農研機構が開発したナスの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/eggplant/flower
```

判別結果候補

disease

- ・健全
- ・うどんこ病
- ・すすかび病
- ・すす病
- ・半身萎凋病
- ・灰色かび病
- ・褐色円星病
- ・褐色腐敗病
- ・青枯病
- ・黒枯病
- ・アザミウマ類
- ・アブラムシ類
- ・オオタバコガ
- ・オオニジュウヤホシテントウ
- ・カメムシ類
- ・コガネムシ類
- ・コナジラミ類
- ・チャノホコリダニ
- ・ハスモンヨトウ
- ・ハダニ類
- ・ハモグリバエ類
- ・ヨトウガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・花

分類検出型病害判別器[ブドウ・葉表裏] (POST naro/rcait-v2/grape/leaf)

農研機構が開発したブドウの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/grape/leaf
```

判別結果候補

disease

- ・健全
- ・べと病
- ・晩腐病
- ・褐斑病
- ・チャノコカクモンハマキ
- ・ハスモンヨトウ
- ・ブドウハモグリダニ
- ・フタテンヒメヨコバイ
- ・モンキクロノメイガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病害判別器[ブドウ・果実] (POST naro/rcait-v2/grape/fruit)

農研機構が開発したブドウの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/grape/fruit
```

判別結果候補

disease

- ・健全
- ・べと病
- ・晩腐病
- ・褐斑病
- ・チャノコカクモンハマキ
- ・ハスモンヨトウ

- ・ブドウハモグリダニ
- ・フタテンヒメヨコバイ
- ・モンキクロノメイガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・果実

分類検出型病虫害判別器[ピーマン・葉表裏] (POST narо/rcait-v2/greenpepper/leaf)

農研機構が開発したピーマンの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/greenpepper/leaf
```

判別結果候補

disease

- ・健全
- ・モザイク病
- ・黄化えそ病
- ・うどんこ病
- ・斑点病
- ・黒枯病
- ・アザミウマ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病虫害判別器[タマネギ・葉表裏] (POST narо/rcait-v2/onion/leaf)

農研機構が開発したタマネギの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/onion/leaf
```

判別結果候補

disease

- ・健全
- ・ハモグリバエ類
- ・黒斑病

- ・アザミウマ類
- ・べと病
- ・ネギコガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病害判別器[モモ・葉表裏] (POST naro/rcait-v2/peach/leaf)

農研機構が開発したモモの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/peach/leaf
```

判別結果候補

disease

- ・健全
- ・せん孔細菌病
- ・縮葉病
- ・ハダニ類
- ・モモハモグリガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病害判別器[モモ・果実] (POST naro/rcait-v2/peach/fruit)

農研機構が開発したモモの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/peach/fruit
```

判別結果候補

disease

- ・健全
- ・せん孔細菌病
- ・縮葉病
- ・ハダニ類

- ・モモハモグリガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・果実

分類検出型病虫害判別器[ジャガイモ・葉表裏] (POST nara/rcait-v2/potato/leaf)

農研機構が開発したジャガイモの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/nara-rcait/pests/plant_disease/nara/rcait-v2/potato/leaf
```

判別結果候補

disease

- ・健全
- ・モザイク病
- ・夏疫病
- ・疫病
- ・オオニジュウヤホシテントウ
- ・ナストビハムシ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病虫害判別器[カボチャ・葉表裏] (POST nara/rcait-v2/pumpkin/leaf)

農研機構が開発したカボチャの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/nara-rcait/pests/plant_disease/nara/rcait-v2/pumpkin/leaf
```

判別結果候補

disease

- ・健全
- ・CMV
- ・WMV
- ・ZYMV
- ・うどんこ病
- ・ウリハムシ

- ・ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病虫害判別器[エダマメ・葉表裏] (POST nara/rcait-v2/soybean/leaf)

農研機構が開発したエダマメの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/nara-rcait/pests/plant_disease/nara/rcait-v2/soybean/leaf
```

判別結果候補

disease

- ・健全
- ・べと病
- ・モザイク病
- ・細菌性病害
- ・褐紋病
- ・黒根腐病
- ・ウコンノメイガ
- ・コガネムシ類
- ・ハスモンヨトウ
- ・ハダニ類
- ・ハモグリバエ類
- ・フタスジヒメハムシ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病虫害判別器[イチゴ・葉表裏] (POST nara/rcait-v2/strawberry/leaf)

農研機構が開発したイチゴの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/nara-rcait/pests/plant_disease/nara/rcait-v2/strawberry/leaf
```

判別結果候補

disease

- ・健全
- ・うどんこ病
- ・灰色かび病
- ・炭疽病
- ・萎黄病
- ・輪斑病
- ・アザミウマ類
- ・アブラムシ類
- ・コナジラミ類
- ・ハスモンヨトウ
- ・ハダニ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病虫害判別器[イチゴ・果実] (POST narо/rcait-v2/strawberry/fruit)

農研機構が開発したイチゴの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/strawberry/fruit
```

判別結果候補

disease

- ・健全
- ・うどんこ病
- ・灰色かび病
- ・炭疽病
- ・萎黄病
- ・輪斑病
- ・アザミウマ類
- ・アブラムシ類
- ・コナジラミ類
- ・ハスモンヨトウ
- ・ハダニ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・果実

分類検出型病害判別器[イチゴ・花] (POST naro/rcait-v2/strawberry/flower)

農研機構が開発したイチゴの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/strawberry/flower
```

判別結果候補

disease

- ・健全
- ・うどんこ病
- ・灰色かび病
- ・炭疽病
- ・萎黄病
- ・輪斑病
- ・アザミウマ類
- ・アブラムシ類
- ・コナジラミ類
- ・ハスモンヨトウ
- ・ハダニ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・花
-

分類検出型病害判別器[トマト・葉表裏] (POST naro/rcait-v2/tomato/leaf)

農研機構が開発したトマトの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/tomato/leaf
```

判別結果候補

disease

- ・健全
- ・モザイク病
- ・うどんこ病
- ・かいよう病
- ・すすかび病
- ・灰色かび病
- ・疫病

- ・葉かび病
- ・褐色輪紋病
- ・青枯病
- ・黄化葉巻病
- ・アザミウマ類
- ・アブラムシ類
- ・オオタバコガ
- ・コナジラミ類
- ・トマトサビダニ
- ・ハスモンヨトウ
- ・ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類検出型病害判別器[トマト・果実] (POST narо/rcait-v2/tomato/fruit)

農研機構が開発したトマトの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/tomato/fruit
```

判別結果候補

disease

- ・健全
- ・モザイク病
- ・うどんこ病
- ・かいよう病
- ・すすかび病
- ・灰色かび病
- ・疫病
- ・葉かび病
- ・褐色輪紋病
- ・青枯病
- ・黄化葉巻病
- ・アザミウマ類
- ・アブラムシ類
- ・オオタバコガ
- ・コナジラミ類
- ・トマトサビダニ
- ・ハスモンヨトウ

- ・ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・果実

分類検出型病害判別器[トマト・花] (POST naro/rcait-v2/tomato/flower)

農研機構が開発したトマトの分類検出型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/tomato/flower
```

判別結果候補

disease

- ・健全
- ・モザイク病
- ・うどんこ病
- ・かいよう病
- ・すすかび病
- ・灰色かび病
- ・疫病
- ・葉かび病
- ・褐色輪紋病
- ・青枯病
- ・黄化葉巻病
- ・アザミウマ類
- ・アブラムシ類
- ・オオタバコガ
- ・コナジラミ類
- ・トマトサビダニ
- ・ハスモンヨトウ
- ・ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・花

4.1.2 現行(2026/03/03リリース)版分類検出型病虫害判別器 - レスポンスボディ

Data Typeは、[ECMA-404 2nd Edition](#)を参考に定義しています。

i レスポンスサンプル**Response JSON**

```
{
  "assets": [
    {
      "id": null,
      "comments": null,
      "NARO_PPD_Request_Id": "xLhRD3iDKhSVRDnoUCQVvj73",
      "images": [
        {
          "filename": "haiirokabibyo.jpg",
          "annotations": [
            {
              "iso8601": "2026-03-01T20:00:28.890302+09:00",
              "type": "ai_hybrid",
              "annotator": "naro v0.1.Y2021DV2 tomato flower classifier",
              "options": null,
              "labels": {
                "disease": [
                  {
                    "class": "すすかび病",
                    "score": 0.8793310513448739
                  },
                  {
                    "class": "灰色かび病",
                    "score": 0.3201711950197813
                  },
                  {
                    "class": "オオタバコガ",
                    "score": 0.036773190211159386
                  },
                  {
                    "class": "褐色輪紋病",
                    "score": 0.003299356529982962
                  },
                  {
                    "class": "トマトサビダニ",
                    "score": 0.0024823041443379355
                  },
                  {
                    "class": "コナジラミ類",
                    "score": 0.00222607805654861
                  },
                  {
                    "class": "ToMV",
                    "score": 0.0012093066636671918
                  },
                  {
                    "class": "黄化葉巻病",
                    "score": 0.0007083310990053263
                  },
                  {
                    "class": "健全",
                    "score": 0.00047738904738309283
                  },
                  {
                    "class": "ハスモンヨトウ",
                    "score": 0.0004187754962257907
                  },
                  {
                    "class": "ハモグリバエ類",
                    "score": 0.00038415188417341027
                  },
                  {
                    "class": "アザミウマ類",
                    "score": 0.0003392109612146271
                  },
                  {
                    "class": "疫病",
                    "score": 0.00012663866692875803
                  },
                  {
                    "class": "葉かび病",
                    "score": 0.00011396089745816058
                  },
                  {
                    "class": "うどんこ病",
                    "score": 0.00007669495803220157
                  },
                  {
                    "class": "かいよう病",
                    "score": 0.00006873673779221025
                  },
                  {
                    "class": "青枯病",
                    "score": 0.00004769866899193328
                  },
                  {
                    "class": "アブラムシ類",
                    "score": 0.000027281825907495543
                  }
                ]
              }
            }
          ]
        }
      ]
    }
  ],
  "boundingBoxes": [
    {

```

```
"bbox": [
  189.12840270996094,
  220.2899627685547,
  222.86680603027344,
  265.23150634765625
],
"score": 0.9554352760314941,
"ratio": [
  1.0,
  1.0
],
"Labels": {
  "disease": [
    {
      "class": "灰色かび病",
      "score": null
    }
  ]
}
},
{
  "bbox": [
    177.40538024902344,
    106.50698852539062,
    218.06143188476562,
    161.90274047851562
  ],
  "score": 0.9114187359809875,
  "ratio": [
    1.0,
    1.0
  ],
  "Labels": {
    "disease": [
      {
        "class": "灰色かび病",
        "score": null
      }
    ]
  }
},
{
  "bbox": [
    222.55186462402344,
    125.38973236083984,
    266.1239013671875,
    160.02926635742188
  ],
  "score": 0.8206420540809631,
  "ratio": [
    1.0,
    1.0
  ],
  "Labels": {
    "disease": [
      {
        "class": "灰色かび病",
        "score": null
      }
    ]
  }
},
{
  "bbox": [
    179.79420471191406,
    123.38438415527344,
    208.10662841796875,
    162.24171447753906
  ],
  "score": 0.7903687357902527,
  "ratio": [
    1.0,
    1.0
  ],
  "Labels": {
    "disease": [
      {
        "class": "灰色かび病",
        "score": null
      }
    ]
  }
},
{
  "bbox": [
    80.5113525390625,
    93.50646209716797,
    109.89910888671875,
    120.6491928100586
  ],
  "score": 0.7670511603355408,
  "ratio": [
    1.0,
    1.0
  ],
  "Labels": {
    "disease": [
      {
        "class": "灰色かび病",
        "score": null
      }
    ]
  }
}
```

```

    }
  ]
},
{
  "bbox": [
    74.67442321777344,
    177.1818389892578,
    185.94473266601562,
    274.0
  ],
  "score": 0.6993276476860046,
  "ratio": [
    1.0,
    1.0
  ],
  "labels": {
    "disease": [
      {
        "class": "灰色かび病",
        "score": null
      }
    ]
  }
},
{
  "bbox": [
    193.30099487304688,
    102.9327392578125,
    217.8828582763672,
    133.1166534423828
  ],
  "score": 0.5571050643920898,
  "ratio": [
    1.0,
    1.0
  ],
  "labels": {
    "disease": [
      {
        "class": "灰色かび病",
        "score": null
      }
    ]
  }
}
],
  "SHA256": "51acabaeb9a2d2e8c9923b520b8852135749a4f5309363bc3d0769a77813d1c1"
}
]
}

```

body (レスポンスボディ)

Parameter	Data Type	Description
assets	array	レスポンス情報が記述されています。 詳細は、 body["assets"][0] をご覧ください。

body["assets"][0]

Parameter	Data Type	Description
id	string	リクエスト時のデータID情報がコピーされ記述されています。
comments	string	リクエスト時のコメント情報がコピーされ記述されています。 body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"]の値が空の場合、リクエスト時のコメント情報の先頭にリクエストされた作目に該当する病虫害が検出されませんでしたので、空のリストを返却しました。; と追記されます。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i] を参照してください。
NARO_PPD_Request_Id	string	本APIによって決定された16文字の画像診断結果ID情報が記述されています。

Parameter	Data Type	Description
images	array	画像情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]</code> を参照してください。

body["assets"][0]["images"][0]

Parameter	Data Type	Description
filename	string	リクエスト時に指定した画像のファイル名情報がコピーされ記述されています。
annotations	array	リクエスト時に指定した判別器が判定したアノテーション結果の情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]</code> を参照してください。
SHA256	string	リクエスト時の画像から生成されたハッシュ値(SHA256)の16進数文字列が記述されています。
data	string	リクエスト時に、レスポンス情報にリクエスト時の画像を添付する設定を有効にしていた場合(<code>attach_image: true</code>), <code>base64</code> エンコードした画像データが記述されています。 有効にしていない場合、記述されません。

body["assets"][0]["images"][0]["annotations"][0]

Parameter	Data Type	Description
iso8601	string	本APIによる画像診断が行われたタイミングの日時情報がISO 8601形式で記述されています。
type	string	判定者のタイプが記述されています。 分類型病虫害判別器にリクエストした場合は <code>ai_classification</code> と記述されます。
annotator	string	判定者情報が記述されています。 リクエスト時に指定した判別器に応じて名称が異なります。
options	object	リクエスト時に指定した追加設定情報がコピーされ記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["options"]</code> を参照してください。
boundingBoxes	array	リクエスト時に指定した判別器が判定した結果のバウンディングボックス情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]</code> を参照してください。
labels	object	リクエスト時に指定した判別器が判定した結果の情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["labels"]</code> を参照してください。

body["assets"][0]["images"][0]["annotations"][0]["options"]

Parameter	Data Type	Description
diagnosis	boolean	AIによる病虫害画像診断が行われたかどうか記述されています。 リクエスト時に指定していなければ記述されていない、もしくは、 <code>true</code> が記述されています。

Parameter	Data Type	Description
attach_image	boolean	レスポンス情報に、リクエスト時の画像が添付されているかどうか記述されています。 リクエスト時に指定していなければ記述されていない、もしくは、falseが記述されています。
response_json_format	string	リクエスト時に設定した判定結果のレスポンスJSONのフォーマットについて記述されています。 リクエスト時に指定していなければ記述されていません。

body["assets"][0]["images"][0]["annotations"][0]["labels"]

Parameter	Data Type	Description
disease	array	リクエスト時に指定した判別器が出力した結果[病虫害名, 病虫害の確度のセット]が確度の降順で記述されています。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["labels"]["disease"][i] を参照してください。
part	array	リクエスト時に指定した判別器が出力した結果[部位名, 部位の確度のセット]が確度の降順で記述されています。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["labels"]["part"][i] を参照してください。

body["assets"][0]["images"][0]["annotations"][0]["labels"]["disease"][i]

Parameter	Data Type	Description
class	string	病虫害名が記述されています。
score	number	病虫害の確度が記述されています。

body["assets"][0]["images"][0]["annotations"][0]["labels"]["part"][i]

Parameter	Data Type	Description
class	string	部位名が記述されています。
score	number	部位の確度が記述されています。

body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]

リクエスト時に指定した判別器で病害判別不可能な画像がポストされた場合、本パラメータの値は空になります。

Parameter	Data Type	Description
bbox	array	リクエスト時に指定した判別器が判定した結果のバウンディングボックスの座標情報が記述されています。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["bbox"][<index>] を参照して下さい。
score	number	リクエスト時に指定した判別器が判定した結果のバウンディングボックスの確度情報が記述されています。 別な表現で言えば、バウンディングボックス内に labels 以下に示す結果が存在する確度を表しています。

Parameter	Data Type	Description
ratio	array	リクエスト時に指定した判別器が判定した結果のバウンディングボックスのスケール情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["ratio"]<index></code> を参照して下さい。
labels	object	リクエスト時に指定した判別器が判定した結果のバウンディングボックスのアノテーションラベル情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["labels"]</code> を参照してください。

`body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["bbox"]<index>`

リクエスト時に指定した判別器が判定した結果のバウンディングボックスの座標情報が配列形式で記述されています。
画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義されています。

<index>	Data Type	Description
0	number	バウンディングボックスの始点ピクセルのx座標値が記述されています。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義されています。
1	number	バウンディングボックスの始点ピクセルのy座標値が記述されています。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義されています。
2	number	バウンディングボックスの終点ピクセルのx座標値が記述されています。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義されています。
3	number	バウンディングボックスの終点ピクセルのy座標値が記述されています。 画像座標系(画像の左上隅画素の中心を原点 (0, 0) とする座標系)で定義されています。

`body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["ratio"]<index>`

リクエスト時に指定した判別器が判定した結果のバウンディングボックスのスケール情報が配列形式で記述されています。
現行(2022/04/28リリース)版では全て [1.0, 1.0] と記述され、`bbox` で記述されたバウンディングボックスの座標情報をそのまま使うことが出来ます。
このパラメータは、今後のアップデートで判別器が追加されたときに、バウンディングボックスのスケール情報を使用する可能性があるため、実装されています。

<index>	Data Type	Description
0	number	バウンディングボックスのx軸におけるスケール係数が記述されています。 現行(2022/04/28リリース)版では、この値は必ず 1.0 になります。
1	number	バウンディングボックスのy軸におけるスケール係数が記述されています。 現行(2022/04/28リリース)版では、この値は必ず 1.0 になります。

body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["labels"]

Parameter	Data Type	Description
disease	array	リクエスト時に指定した判別器が判定した結果の病害情報が記述されています。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["labels"]["disease"][j] を参照してください。

body["assets"][0]["images"][0]["annotations"][0]["boundingBoxes"][i]["labels"]["disease"][j]

Parameter	Data Type	Description
class	string	リクエスト時に指定した判別器が判定した結果の病害名情報が記述されています。
score	number	リクエスト時に指定した判別器が判定した結果の病害の確度情報が記述されています。 現行(2022/04/28リリース)版の検出型病害判別器では、この値は必ず null となります。 このパラメータは、同じ検出型である検出型虫害判別器とJSONフォーマットを統一するために実装されています。

4.1.3 現行(2026/03/03リリース)版分類型病虫害判別器 - サンプルコード

本サンプルコードは、作成例として提示するものであり、完全な動作を保証しているものではありません。

本サンプルコードは、使用者の責任において、使用（複製、改変を含む）することができます。

本サンプルコードの使用により損害が生じた場合、本機構・本組織はその損害について一切の責を負いません。

本サンプルコードは、Pythonで記述されています。

本サンプルコードでは、Python標準ライブラリ以外に、Requestsライブラリ(<https://requests-docs-jp.readthedocs.io/en/latest/>)を使用しています。

ソースコード

Python

```
from typing import List, Dict, Any, Optional, Union, Tuple
import json
import base64

import requests

def main() -> None:
    # リクエストヘッダーの定義
    request_header: Dict[str, str] = {
        "X-User-Id": "<This parameter is required.>",
        "Authorization": "<This parameter is required.>",
        "X-Authorization": "<This parameter is required.>",
        "Content-Type": "application/json",
    }

    # リクエストボディの定義
    request_body: Dict[str, List[Dict[str, Any]]] = {
        "assets": [
            {
                "id": "<This parameter is not required.>",
                "X-User-Id": "<This parameter is not required.>",
                "comments": "<This parameter is not required.>",
                "options": {
                    "diagnosis": True,
                    "attach_image": False
                },
                "images": [
                    {
                        "filename": "<This parameter is not required.>",
                        "filedate": "<This parameter is not required.>",
                        "data": base64.b64encode(open("test.jpg", "rb").read()).decode("utf-8")
                    }
                ]
            }
        ]
    }

    # サーバーへポスト
    response = requests.post(
        url="https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v2/cucumber/leaf",
        headers=request_header,
        data=json.dumps(request_body)
    )

    # レスポンスボディの取得
    response_body: Dict[str, List[Dict[str, Any]]] = json.loads(response.content.decode('utf-8'))

    # レスポンスボディの表示
    print(json.dumps(response_body, indent=4, ensure_ascii=False))

main()
```

結果

Response JSON

```
{
  "assets": [
    {
      "id": "<This parameter is not required.>",
      "comments": "<This parameter is not required.>",
      "NARO_PPD_Request_Id": "qt270cuvYf014csK",
      "images": [
        {
          "filename": "<This parameter is not required.>",
          "annotations": [
            {
              "iso8601": "2026-03-01T20:04:39.292622+09:00",

```

```
"type": "ai_hybrid",
"annotator": "naro v0.1.V2021Dv2 cucumber leaf classifier",
"options": null,
"Labels": {
  "disease": [
    {
      "class": "炭疽病",
      "score": 0.17059489691116483
    },
    {
      "class": "チャノホコリダニ",
      "score": 0.005625869042085048
    },
    {
      "class": "褐斑病",
      "score": 0.004024475431232501
    },
    {
      "class": "灰色かび病",
      "score": 0.0022632188128612163
    },
    {
      "class": "CMV",
      "score": 0.0006697244161748215
    },
    {
      "class": "CCVV",
      "score": 0.00021184309818717025
    },
    {
      "class": "オオタバコガ",
      "score": 0.00020412817163179342
    },
    {
      "class": "コナジラミ類",
      "score": 0.00019714444999440565
    },
    {
      "class": "MYSV",
      "score": 0.00012935418691334798
    },
    {
      "class": "ハモグリバエ類",
      "score": 0.00011943200462402888
    },
    {
      "class": "すす病",
      "score": 0.00011264795416245772
    },
    {
      "class": "うどんこ病",
      "score": 0.00008860059841878258
    },
    {
      "class": "アザミウマ類",
      "score": 0.00008768222068968802
    },
    {
      "class": "WMV",
      "score": 0.00007449971292557715
    },
    {
      "class": "つる枯病",
      "score": 0.0000716321837976992
    },
    {
      "class": "ウリキンウワバ",
      "score": 0.00006705156580493186
    },
    {
      "class": "ハスモンヨトウ",
      "score": 0.00004452123747108196
    },
    {
      "class": "べと病",
      "score": 0.000030337389053436153
    },
    {
      "class": "健全",
      "score": 0.0000183079298321877
    },
    {
      "class": "斑点細菌病",
      "score": 0.000016980040065473153
    },
    {
      "class": "ZYMV",
      "score": 0.000014866491707060435
    },
    {
      "class": "ハダニ類",
      "score": 0.00001453953135013856
    }
  ]
}
```

```
        "class": "アブラムシ類",
        "score": 0.0000035134891854513775
      }
    ],
    "boundingBoxes": [
      {
        "bbox": [
          218.1205291748047,
          122.42230987548828,
          270.37677001953125,
          161.06893920898438
        ],
        "score": 0.8625728487968445,
        "ratio": [
          1.0,
          1.0
        ],
        "labels": {
          "disease": [
            {
              "class": "炭疽病",
              "score": null
            }
          ]
        }
      }
    ]
  },
  "SHA256": "51acabaeb9a2d2e8c9923b520b8852135749a4f5309363bc3d0769a77813d1c1"
}
}
```

5. 現行(2022/04/28リリース)版判別器

5.1 分類型病虫害判別器

5.1.1 現行(2022/04/28リリース)版分類型病虫害判別器

分類型病害判別器では、POSTされた画像に写っている病虫害の分類を行い、その結果を返します。

また、同一URLのGETメソッドを用いることで、判定結果候補となる病害のリストを返します。現行(2022/04/28リリース)版分類型病虫害判別器は、以下の判別器のことを指します。「部位指定なし」と表記されている場合は、病虫害だけでなく、部位情報も返します。

- ・分類型病虫害判別器[キク・部位指定なし] (POST naro/rcait-v1/chrysanthemum)
- ・分類型病虫害判別器[ブドウ・部位指定なし] (POST naro/rcait-v1/grape)
- ・分類型病虫害判別器[ピーマン・部位指定なし] (POST naro/rcait-v1/greenpepper)
- ・分類型病虫害判別器[タマネギ・部位指定なし] (POST naro/rcait-v1/onion)
- ・分類型病虫害判別器[モモ・部位指定なし] (POST naro/rcait-v1/peach)
- ・分類型病虫害判別器[ジャガイモ・部位指定なし] (POST naro/rcait-v1/potato)
- ・分類型病虫害判別器[カボチャ・部位指定なし] (POST naro/rcait-v1/pumpkin)
- ・分類型病虫害判別器[ダイズ・部位指定なし] (POST naro/rcait-v1/soybean)

リクエスト

ヘッダー

[共通リクエストヘッダー](#)を参照してください。

ボディ

[共通リクエストボディ](#)を参照してください。

レスポンス

ステータスコード

成功時

200

エラー発生時

[エラー発生時のレスポンス](#)を参照してください。

ボディ

AIによる病虫害画像診断を行う場合

[現行\(2022/04/28リリース\)版分類型病虫害判別器 - レスポンスボディ](#)を参照してください。

人によるアノテーション情報を登録する場合

[人によるアノテーション情報を登録する場合 - レスポンスボディ](#)を参照してください。

サンプルコード

AIによる病虫害画像診断を行う場合

[現行\(2022/04/28リリース\)版分類型病虫害判別器 - サンプルコード](#)をご覧ください。

人によるアノテーション情報を登録する場合

[人によるアノテーション情報を登録する場合 - サンプルコードをご覧ください。](#)

分類型病虫害判別器[キク・部位指定なし] (POST nro/rcait-v1/chrysanthemum)

農研機構が開発したキクの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/nro-rcait/pests/plant_disease/nro/rcait-v1/chrysanthemum
```

判別結果候補

disease

- ・ミカンキイロアザミウマ
- ・健全
- ・えそ病 (TSWV)
- ・白さび病

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉 (表)
- ・葉 (裏)

分類型病虫害判別器[ブドウ・部位指定なし] (POST nro/rcait-v1/grape)

農研機構が開発したブドウの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/nro-rcait/pests/plant_disease/nro/rcait-v1/grape
```

判別結果候補

disease

- ・健全
- ・根頭がんしゅ病
- ・ハスモンヨトウ
- ・フタテンヒメヨコバイ
- ・奇形葉 (生理障害等)
- ・生理障害 (マグネシウム欠乏症)
- ・べと病
- ・褐斑病

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・主幹部（成木）
- ・葉（表）
- ・葉（裏）

分類型病虫害判別器[ピーマン・部位指定なし] (POST naro/rcait-v1/greenpepper)

農研機構が開発したピーマンの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/greenpepper
```

判別結果候補

disease

- ・健全
- ・斑点病
- ・うどんこ病
- ・モザイク病（PMMoV）
- ・えそ病（TSWV）
- ・黒枯病
- ・ミカンキイロアザミウマ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類型病虫害判別器[タマネギ・部位指定なし] (POST naro/rcait-v1/onion)

農研機構が開発したタマネギの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/onion
```

判別結果候補

disease

- ・ネギコガ
- ・ネギアザミウマ
- ・ネギハモグリバエ
- ・健全
- ・べと病

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・葉（裏）

分類型病虫害判別器[モモ・部位指定なし] (POST naro/rcait-v1/peach)

農研機構が開発したモモの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/peach
```

判別結果候補

disease

- ・健全
- ・せん孔細菌病
- ・カンザワハダニ
- ・生理障害
- ・葉害（葉斑を含む）

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉（表）
- ・果実（幼果期）
- ・果実（成熟期～収穫期）
- ・枝

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

分類型病虫害判別器[ジャガイモ・部位指定なし] (POST naro/rcait-v1/potato)

農研機構が開発したジャガイモの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/potato
```

判別結果候補

disease

- ・ナストビハムシ
- ・オオニジュウヤホシテントウ
- ・健全
- ・夏疫病
- ・疫病

- ・モザイク病 (PVY)
- ・モザイク病 (PVY) えそ症状

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉 (表)

分類型病虫害判別器[カボチャ・部位指定なし] (POST naro/rcait-v1/pumpkin)

農研機構が開発したカボチャの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/pumpkin
```

判別結果候補

disease

- ・ウリハムシ
- ・健全
- ・うどんこ病
- ・モザイク病 (CMV)
- ・モザイク病 (WMV)
- ・モザイク病 (ZYMV)
- ・ハモグリバエ類

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- ・葉 (表)

分類型病虫害判別器[ダイズ・部位指定なし] (POST naro/rcait-v1/soybean)

農研機構が開発したダイズの分類型病害判別器を使用するためのAPIです。

URL (APIの絶対パス)

```
https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/soybean
```

判別結果候補

disease

- ・ハスモンヨトウ
- ・フタスジヒメハムシ
- ・ダイズクロハモグリバエ
- ・健全
- ・べと病

- モザイク病
- 細菌性病害
- 褐紋病
- ハダニ類
- コガネムシ類
- 黒根腐病
- ウコンノメイガ

判別結果候補のリストは同一URLでGETメソッドを使用することで獲得できます。

part

- 葉 (表)

5.1.2 現行(2022/04/28リリース)版分類型病虫害判別器 - レスポンスボディ

Data Typeは、[ECMA-404 2nd Edition](#)を参考に定義しています。

レスポンスサンプル

Response JSON

```
{
  "assets": [
    {
      "id": "<This parameter is not required.>",
      "comments": "<This parameter is not required.>",
      "NARO_PPD_Request_Id": "qt270cuvf014cSk",
      "images": [
        {
          "filename": "<This parameter is not required.>",
          "annotations": [
            {
              "iso8601": "2022-11-01T15:59:23.248501+09:00",
              "type": "ai_classification",
              "annotator": "naro v1 chrysanthemum classifier",
              "options": {
                "diagnosis": true,
                "attach_image": false
              },
              "labels": {
                "disease": [
                  {
                    "class": "白さび病",
                    "score": 0.9998755557173922
                  },
                  {
                    "class": "えそ病 (TSW)",
                    "score": 7.80839511458524e-05
                  },
                  {
                    "class": "健全",
                    "score": 4.291442594279759e-05
                  },
                  {
                    "class": "ミカンキイロアザミウマ",
                    "score": 3.4459055191886517e-06
                  }
                ]
              },
              "part": [
                {
                  "class": "葉 (表)",
                  "score": 0.9999990853449507
                },
                {
                  "class": "葉 (裏)",
                  "score": 9.146550492191732e-07
                }
              ]
            }
          ]
        }
      ],
      "SHA256": "b5b57f43fac73505e1c7a38a85d9bd5f7c6b1170f526196b80c84d69066039d1"
    }
  ]
}
```

body (レスポンスボディ)

Parameter	Data Type	Description
assets	array	レスポンス情報が記述されています。 詳細は、 body["assets"][0] をご覧ください。

body["assets"][0]

Parameter	Data Type	Description
id	string	リクエスト時のデータID情報がコピーされ記述されています。

Parameter	Data Type	Description
comments	string	リクエスト時のコメント情報がコピーされ記述されています。
NARO_PPD_Request_Id	string	本APIによって決定された16文字の画像診断結果ID情報が記述されています。
images	array	画像情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]</code> を参照してください。

body["assets"][0]["images"][0]

Parameter	Data Type	Description
filename	string	リクエスト時に指定した画像のファイル名情報がコピーされ記述されています。
annotations	array	リクエスト時に指定した判別器が判定したアノテーション結果の情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]</code> を参照してください。
SHA256	string	リクエスト時の画像から生成されたハッシュ値(SHA256)の16進数文字列が記述されています。
data	string	リクエスト時に、レスポンス情報にリクエスト時の画像を添付する設定を有効にしていた場合(<code>attach_image: true</code>), <code>base64</code> エンコードした画像データが記述されています。 有効にしていない場合、記述されません。

body["assets"][0]["images"][0]["annotations"][0]

Parameter	Data Type	Description
iso8601	string	本APIによる画像診断が行われたタイミングの日時情報がISO 8601形式で記述されています。
type	string	判定者のタイプが記述されています。 分類型病虫害判別器にリクエストした場合は <code>ai_classification</code> と記述されます。
annotator	string	判定者情報が記述されています。 リクエスト時に指定した判別器に応じて名称が異なります。
options	object	リクエスト時に指定した追加設定情報がコピーされ記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["options"]</code> を参照してください。
labels	object	リクエスト時に指定した判別器が判定した結果の情報が記述されています。 詳細は、 <code>body["assets"][0]["images"][0]["annotations"][0]["labels"]</code> を参照してください。

body["assets"][0]["images"][0]["annotations"][0]["options"]

Parameter	Data Type	Description
diagnosis	boolean	AIによる病虫害画像診断が行われたかどうか記述されています。 リクエスト時に指定していなければ記述されていない、もしくは、 <code>true</code> が記述されています。

Parameter	Data Type	Description
attach_image	boolean	レスポンス情報に、リクエスト時の画像が添付されているかどうか記述されています。 リクエスト時に指定していなければ記述されていない、もしくは、falseが記述されています。

body["assets"][0]["images"][0]["annotations"][0]["labels"]

Parameter	Data Type	Description
disease	array	リクエスト時に指定した判別器が出力した結果[病虫害名, 病虫害の確度のセット]が確度の降順で記述されています。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["labels"]["disease"][i] を参照してください。
part	array	リクエスト時に指定した判別器が出力した結果[部位名, 部位の確度のセット]が確度の降順で記述されています。 詳細は、 body["assets"][0]["images"][0]["annotations"][0]["labels"]["part"][i] を参照してください。

body["assets"][0]["images"][0]["annotations"][0]["labels"]["disease"][i]

Parameter	Data Type	Description
class	string	病虫害名が記述されています。
score	number	病虫害の確度が記述されています。

body["assets"][0]["images"][0]["annotations"][0]["labels"]["part"][i]

Parameter	Data Type	Description
class	string	部位名が記述されています。
score	number	部位の確度が記述されています。

5.1.3 現行(2022/04/28リリース)版分類型病虫害判別器 - サンプルコード

本サンプルコードは、作成例として提示するものであり、完全な動作を保証しているものではありません。

本サンプルコードは、使用者の責任において、使用（複製、改変を含む）することができます。

本サンプルコードの使用により損害が生じた場合、本機構・本組織はその損害について一切の責を負いません。

本サンプルコードは、Pythonで記述されています。

本サンプルコードでは、Python標準ライブラリ以外に、Requestsライブラリ(<https://requests-docs-jp.readthedocs.io/en/latest/>)を使用しています。

ソースコード

Python

```
from typing import List, Dict, Any, Optional, Union, Tuple
import json
import base64

import requests

def main() -> None:
    # リクエストヘッダーの定義
    request_header: Dict[str, str] = {
        "X-User-Id": "<This parameter is required.>",
        "Authorization": "<This parameter is required.>",
        "X-Authorization": "<This parameter is required.>",
        "Content-Type": "application/json",
    }

    # リクエストボディの定義
    request_body: Dict[str, List[Dict[str, Any]]] = {
        "assets": [
            {
                "id": "<This parameter is not required.>",
                "X-User-Id": "<This parameter is not required.>",
                "comments": "<This parameter is not required.>",
                "options": {
                    "diagnosis": True,
                    "attach_image": False
                },
                "images": [
                    {
                        "filename": "<This parameter is not required.>",
                        "filedate": "<This parameter is not required.>",
                        "data": base64.b64encode(open("test.jpg", "rb").read()).decode("utf-8")
                    }
                ]
            }
        ]
    }

    # サーバーへポスト
    response = requests.post(
        url="https://api.wagri2.net/naro-rcait/pests/plant_disease/naro/rcait-v1/chrysanthemum",
        headers=request_header,
        data=json.dumps(request_body)
    )

    # レスポンスボディの取得
    response_body: Dict[str, List[Dict[str, Any]]] = json.loads(response.content.decode('utf-8'))

    # レスポンスボディの表示
    print(json.dumps(response_body, indent=4, ensure_ascii=False))

main()
```

結果

Response JSON

```
{
  "assets": [
    {
      "id": "<This parameter is not required.>",
      "comments": "<This parameter is not required.>",
      "NARO_PPD_Request_Id": "qt270cuvYf014csK",
      "images": [
        {
          "filename": "<This parameter is not required.>",
          "annotations": [
            {
              "iso8601": "2022-11-01T15:59:23.248501+09:00",
            }
          ]
        }
      ]
    }
  ]
}
```

```
"type": "ai_classification",
"annotator": "naro v1 chrysanthemum classifier",
"options": {
  "diagnosis": true,
  "attach_image": false
},
"Labels": {
  "disease": [
    {
      "class": "白さび病",
      "score": 0.9998755557173922
    },
    {
      "class": "えそ病 (TSW)",
      "score": 7.80839511458524e-05
    },
    {
      "class": "健全",
      "score": 4.291442594279759e-05
    },
    {
      "class": "ミカンキイロアザミウマ",
      "score": 3.4459055191886517e-06
    }
  ],
  "part": [
    {
      "class": "葉 (表)",
      "score": 0.999990853449507
    },
    {
      "class": "葉 (裏)",
      "score": 9.146550492191732e-07
    }
  ]
},
"SHA256": "b5b57f43fac73505e1c7a38a85d9bd5f7c6b1170f526196b80c84d69066039d1"
}
```

5.2 人によるアノテーション情報を登録する場合

5.2.1 人によるアノテーション情報を登録する場合 - レスポンスボディ

Data Typeは、[ECMA-404 2nd Edition](#)を参考に定義しています。

body (レスポンスボディ)

Parameter	Data Type	Description
assets	array	レスポンス情報が記述されています。 詳細は、 <code>body["assets"][0]</code> をご覧ください。

body["assets"][0]

Parameter	Data Type	Description
NARO_PPD_Request_Id	string	本APIによって決定された16文字の画像診断結果ID情報が記述されています。
SHA256	string	リクエスト時の画像から生成されたハッシュ値(SHA256)の16進数文字列が記述されています。
comments	string	リクエスト時のコメント情報がコピーされ記述されています。 また、人によるアノテーション情報を登録する(AIによる病虫害画像診断を行わない)場合は、どのAPIのエンドポイントにリクエストしたかの情報も追記されます。
id	string	リクエスト時のデータID情報がコピーされ記述されています。
iso8601	string	本APIによる画像診断が行われたタイミングの日時情報がISO 8601形式で記述されています。

5.2.2 人によるアノテーション情報を登録する場合 - サンプルコード

本サンプルコードは、作成例として提示するものであり、完全な動作を保証しているものではありません。

本サンプルコードは、使用者の責任において、使用（複製、改変を含む）することができます。

本サンプルコードの使用により損害が生じた場合、本機構・本組織はその損害について一切の責を負いません。

本サンプルコードは、Pythonで記述されています。

本サンプルコードでは、Python標準ライブラリ以外に、Requestsライブラリ(<https://requests-docs-jp.readthedocs.io/en/latest/>)を使用しています。

画像と作目名・病害名のアノテーションを行う場合

ソースコード

Python

```
from typing import List, Dict, Any, Optional, Union, Tuple
import json
import base64

import requests

def main() -> None:
    # リクエストヘッダーの定義
    request_header: Dict[str, str] = {
        "X-User-Id": "<This parameter is required.>",
        "Authorization": "<This parameter is required.>",
        "X-Authorization": "<This parameter is required.>",
        "Content-Type": "application/json",
    }

    # リクエストボディの定義
    request_body: Dict[str, List[Dict[str, Any]]] = {
        "assets": [
            {
                "id": "<This parameter is not required.>",
                "X-User-Id": "<This parameter is not required.>",
                "comments": "<This parameter is not required.>",
                "options": {
                    "diagnosis": False,
                    "attach_image": False
                },
            },
        ],
        "images": [
            {
                "filename": "<This parameter is not required.>",
                "filedate": "<This parameter is not required.>",
                "data": base64.b64encode(open("test.png", "rb").read()).decode("utf-8"),
                "annotations": [
                    {
                        "type": "human_classification",
                        "annotator": "<This parameter is not required.>",
                        "Labels": [
                            {
                                "plant": "cucumber",
                                "class": "斑点細菌病",
                                "score": 9999,
                                "place": "36.02790370512755, 140.10198705396417"
                            }
                        ]
                    }
                ]
            }
        ]
    }

    # サーバーへポスト
    response = requests.post(
        url="https://api.wagri2.net/naro-rcait/pests/plant_disease/hosei/hosei-v1/cucumber/leaf-abaxial",
        headers=request_header,
        data=json.dumps(request_body)
    )

    # レスポンスボディの取得
    response_body: Dict[str, List[Dict[str, Any]]] = json.loads(response.content.decode('utf-8'))

    # レスポンスボディの表示
    print(json.dumps(response_body, indent=4, ensure_ascii=False))
```

```
main()
```

結果

Response JSON

```
{
  "assets": [
    {
      "NARO_PPD_Request_Id": "lkja1XkrYyMocIR8",
      "SHA256": "f538418e6f14a6d8b8cc96b646db951797026e464f0102924d89bb4a33db416e",
      "comments": "ENGINE URL: </plant_disease/hosei/hosei-v1/cucumber/leaf-abaxial> .",
      "id": "<This parameter is not required.>",
      "iso8601": "2022-10-31T13:38:46.909887+09:00"
    }
  ]
}
```

画像と作目名・病害名・バウンディングボックスのアノテーションを行う場合

ソースコード

Python

```
from typing import List, Dict, Any, Optional, Union, Tuple
import json
import base64

import requests

def main() -> None:
    # リクエストヘッダーの定義
    request_header: Dict[str, str] = {
        "X-User-Id": "<This parameter is required.>",
        "Authorization": "<This parameter is required.>",
        "X-Authorization": "<This parameter is required.>",
        "Content-Type": "application/json",
    }

    # リクエストボディの定義
    request_body: Dict[str, List[Dict[str, Any]]] = {
        "assets": [
            {
                "id": "<This parameter is not required.>",
                "X-User-Id": "<This parameter is not required.>",
                "comments": "<This parameter is not required.>",
                "options": {
                    "diagnosis": False,
                    "attach_image": False
                },
            },
            {
                "images": [
                    {
                        "filename": "<This parameter is not required.>",
                        "filedate": "<This parameter is not required.>",
                        "data": base64.b64encode(open("test.jpg", "rb").read()).decode("utf-8"),
                        "annotations": [
                            {
                                "type": "human_classification",
                                "annotator": "<annotator name>",
                                "label": [
                                    {
                                        "plant": "cucumber",
                                        "class": "斑点細菌病",
                                        "score": 9999,
                                        "place": "36.02790370512755, 140.10198705396417",
                                        "boundingBoxes": [
                                            {
                                                "bbox": [
                                                    660,
                                                    685,
                                                    721,
                                                    771
                                                ],
                                                "byhand": True,
                                                "comment": "<Comment>"
                                            }
                                        ]
                                    }
                                ]
                            }
                        ]
                    }
                ]
            }
        ]
    }
```

```
]
}

# サーバーへポスト
response = requests.post(
    url="https://api.wagri2.net/naro-rcait/pests/plant_disease/hosei/hosei-v5/cucumber/leaf-adaxial",
    headers=request_header,
    data=json.dumps(request_body)
)

# レスポンスボディの取得
response_body: Dict[str, List[Dict[str, Any]]] = json.loads(response.content.decode('utf-8'))

# レスポンスボディの表示
print(json.dumps(response_body, indent=4, ensure_ascii=False))

main()
```

結果

Response JSON

```
{
  "assets": [
    {
      "NARO_PPD_Request_Id": "qt270cuv9HI5H8ig",
      "SHA256": "4509d3642d6807968027e5a718ef8c9bdf4266e3bf4a12cdd611c895f7cc3f1d",
      "comments": "ENGINE URL: </plant_disease/hosei/hosei-v5/cucumber/leaf-adaxial> .",
      "id": "<This parameter is not required.>",
      "iso8601": "2022-11-01T15:02:58.391768+09:00"
    }
  ]
}
```

6. エラー発生時のレスポンス

6.1 エラー発生時のレスポンス

400 Bad Request

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。

401 Unauthorized

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。
result	string	エラーコードが記述されています。

403 Forbidden

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。
result	string	エラーコードが記述されています。

404 Not Found

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。

422 Unprocessable Entity

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。

429 Too Many Requests

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。
result	string	エラーコードが記述されています。

500 Internal Server Error

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。

503 Service Unavailable

BODY (レスポンスボディ)

Parameter	Data Type	Description
message	string	エラーメッセージが記述されています。