Detect text in images

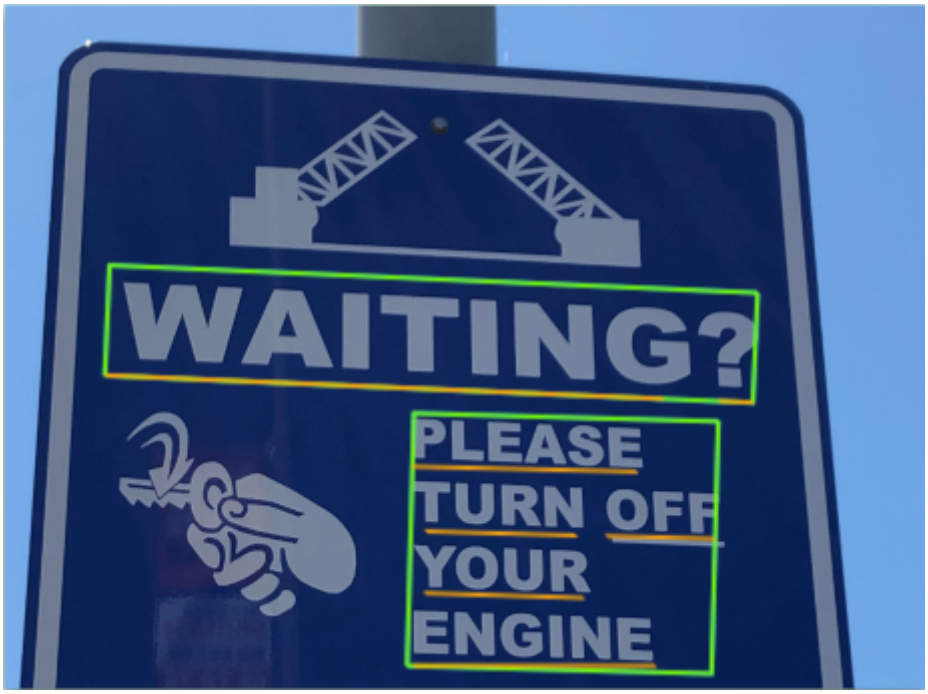
Using this API in a mobile app? Try ML Kit for Firebase (https://firebase.google.com/docs/ml-kit/?hl=zh-tw), which provides native Android and iOS SDKs for using Cloud Vision services, as well as on-device ML Vision APIs and on-device inference using custom ML models.

Optical Character Recognition (OCR)

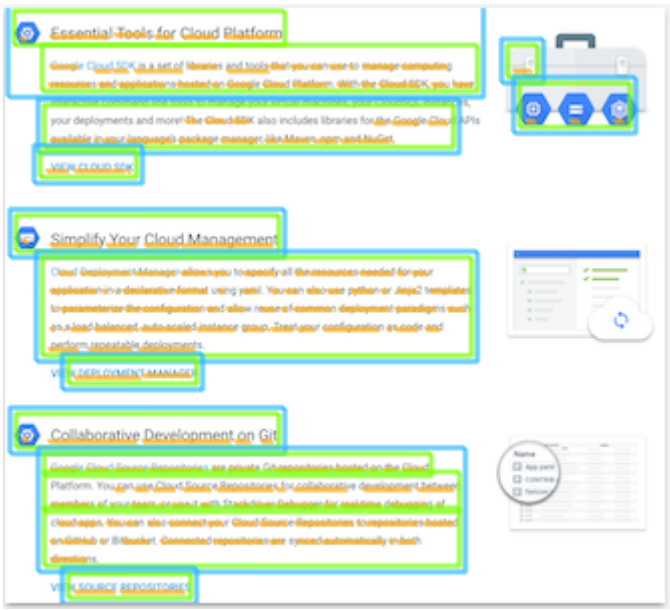
Note: Cloud Vision now supports offline **asynchronous batch image annotation** for all features. This asynchronous request supports up to 2000 image files and returns response JSON files that are stored in your Google Cloud Storage bucket. For more information about this feature, refer to Offline batch image annotation (https://cloud.google.com/vision/docs/batch?hl=zh-tw).

The Vision API can detect and extract text from images. There are ***two*** annotation features that support optical character recognition (OCR):

- `TEXT_DETECTION` detects and **extracts text from any image**. For example, a photograph might contain a street sign or traffic sign. The JSON includes the entire extracted string, as well as individual words, and their bounding boxes.



- `DOCUMENT_TEXT_DETECTION` also extracts text from an image, but the response is **optimized for dense text and documents**. The JSON includes page, block, paragraph, word, and break information.



Learn more about `DOCUMENT_TEXT_DETECTION` for handwriting extraction (https://cloud.google.com/vision/docs/handwriting?hl=zh-tw) and text extraction from files (PDF/TIFF) (https://cloud.google.com/vision/docs/pdf?hl=zh-tw).

Text detection requests

Set up your GCP project and authentication

If you have not created a Google Cloud Platform (GCP) project (https://cloud.google.com/docs/overview/?hl=zh-tw#projects) and service account credentials, do so now. Expand this section for instructions.

1. 設定 GCP Console 專案。

   設定專案

   按一下即可：
   - 建立或選取專案。
   - 啟用該專案Cloud Vision API。
   - 建立服務帳戶。
   - 下載 JSON 格式的私密金鑰。

   您隨時都能在 GCP Console (https://console.cloud.google.com/?hl=zh-tw) 中查看及管理這些資源。

2. 將環境變數 GOOGLE_APPLICATION_CREDENTIALS 設為包含服務帳戶金鑰的 JSON 檔案路徑。 此變數僅適用於您目前的殼層工作階段，所以如果您開啟新的工作階段，請再次設定變數。

   範例： Linux 或 macOS

   將 [PATH] 改成包含您的服務帳戶金鑰的 JSON 檔案路徑。

   ```
   export GOOGLE_APPLICATION_CREDENTIALS="[PATH]"
   ```

   例如：

   ```
   export GOOGLE_APPLICATION_CREDENTIALS="/home/user/Downloads/servi
   ```

   範例： Windows

   將 [PATH] 改成包含您的服務帳戶金鑰的 JSON 檔案路徑，並將 [FILE_NAME] 改成檔案名稱。

   使用 PowerShell：

   ```
   $env:GOOGLE_APPLICATION_CREDENTIALS="[PATH]"
   ```

   例如：

   ```
   $env:GOOGLE_APPLICATION_CREDENTIALS="C:\Users\username\Downloads\
   ```

使用命令提示字元：

```
set GOOGLE_APPLICATION_CREDENTIALS=[PATH]
```

3. 安裝並初始化 Cloud SDK (https://cloud.google.com/sdk/docs/?hl=zh-tw)。

Detect text in a local image

The Vision API can perform feature detection on a local image file by sending the contents of the image file as a base64 encoded (https://cloud.google.com/vision/docs/base64?hl=zh-tw) string in the body of your request.

REST & CMD LINE | C# | GO | JAVA | NODE.JS | PHP | PYTH

Before trying this sample, follow the Python setup instructions in the Vision API Quickstart Using Client Libraries (https://cloud.google.com/vision/docs/quickstart-client-libraries?hl=zh-tw). For more information, see the Vision API Python API reference documentation (https://googleapis.github.io/google-cloud-python/latest/vision/index.html).

TFORM/PYTHON-DOCS-SAMPLES/BLOB/MASTER/VISION/CLOUD-CLIENT/DETECT/DETECT.PY)

FEEDBACK (#)

```
def detect_text(path):
    """Detects text in the file."""
    from google.cloud import vision
    import io
    client = vision.ImageAnnotatorClient()

    with io.open(path, 'rb') as image_file:
        content = image_file.read()

    image = vision.types.Image(content=content)

    response = client.text_detection(image=image)
    texts = response.text_annotations
    print('Texts:')

    for text in texts:
        print('\n"{}"'.format(text.description))

        vertices = (['({},{})'.format(vertex.x, vertex.y)
                     for vertex in text.bounding_poly.vertices])
```

```
        print('bounds: {}'.format(','.join(vertices)))
```

Detect text in a remote image

For your convenience, the Vision API can perform feature detection directly on an image file located in Google Cloud Storage or on the Web without the need to send the contents of the image file in the body of your request.

Caution: When fetching images from HTTP/HTTPS URLs, Google cannot guarantee that the request will be completed. Your request may fail if the specified host denies the request (for example, due to request throttling or DOS (https://en.wikipedia.org/wiki/Denial-of-service_attack) prevention), or if Google throttles requests to the site for abuse prevention. **You should not depend on externally-hosted images for production applications**.

REST & CMD LINE C# GO JAVA NODE.JS PHP 更多選項 PYTH

Before trying this sample, follow the Python setup instructions in the Vision API Quickstart Using Client Libraries (https://cloud.google.com/vision/docs/quickstart-client-libraries?hl=zh-tw). For more information, see the Vision API Python API reference documentation (https://googleapis.github.io/google-cloud-python/latest/vision/index.html).

TFORM/PYTHON-DOCS-SAMPLES/BLOB/MASTER/VISION/CLOUD-CLIENT/DETECT/DETECT.PY)

FEEDBACK (#)

```
def detect_text_uri(uri):
    """Detects text in the file located in Google Cloud Storage or on t
    """
    from google.cloud import vision
    client = vision.ImageAnnotatorClient()
    image = vision.types.Image()
    image.source.image_uri = uri

    response = client.text_detection(image=image)
    texts = response.text_annotations
    print('Texts:')

    for text in texts:
        print('\n"{}"'.format(text.description))
```

```
        vertices = (['({},{})'.format(vertex.x, vertex.y)
                     for vertex in text.bounding_poly.vertices])

        print('bounds: {}'.format(','.join(vertices)))
```

Specify the language (optional)

Both types of OCR requests support one or more **`languageHints`** that specify the language of any text in the image. However, in most cases, an empty value yields the best results since it enables automatic language detection. For languages based on the Latin alphabet, setting **`languageHints`** is not needed. In rare cases, when the language of the text in the image is known, setting a hint helps get better results (although it can be a significant hindrance if the hint is wrong). Text detection returns an error if one or more of the specified languages is not one of the supported languages (https://cloud.google.com/vision/docs/languages?hl=zh-tw).

If you choose to provide a language hint, modify the body of your request (**`request.json`** file) to provide the string of one of the supported languages in the **`imageContext.languageHints`** field as shown below:

```
{
  "requests": [
    {
      "image": {
        "source": {
          "imageUri": "image-url"
        }
      },
      "features": [
        {
          "type": "DOCUMENT_TEXT_DETECTION"
        }
      ],
      "imageContext": {
        "languageHints": ["en-t-i0-handwrit"]
      }
    }
  ]
}
```

How do language hints work?

The `languageHint` format follows the BCP47 (https://tools.ietf.org/html/bcp47) language code formatting guidelines. The BCP47 specified format is as follows:

`language` ["-" `script`] ["-" `region`] *("-" `variant`) *("-" `extension`) ["-" `privateuse`].

For example, the language hint "`en-t-i0-handwrit`" specifies English language (`en`), transform (https://tools.ietf.org/html/rfc6497) extension singleton (`t`), input method engine transform (http://www.unicode.org/reports/tr35/) extension code (`i0`), and handwriting transform code (http://unicode.org/repos/cldr/tags/latest/common/bcp47/transform_ime.xml) (`handwrit`). This roughly says the language is "English transformed from handwriting." You do not need to specify a script code because `Latn` is implied by the "`en`" language.

Multi-regional support

This functionality currently only applies to the OCR feature (types `TEXT_DETECTION` or `DOCUMENT_TEXT_DETECTION`).

You can now specify continent-level data storage and OCR processing. The following regions are currently supported:

- `us`: USA country only
- `eu`: The European Union

Locations

Cloud Vision offers you some control over where the resources for your project are stored and processed. In particular, you can configure Cloud Vision to store and process your data only in the European Union.

By default Cloud Vision stores and processes resources in a **Global** location, which means that Cloud Vision doesn't guarantee that your resources will remain within a particular location or region. If you choose the **European Union** location, Google will store your data and process it only in the European Union. You and your users can access the data from any location.

Setting the location using the API

Cloud Vision supports both a global API endpoint (`vision.googleapis.com`) and a European Union endpoint (`eu-vision.googleapis.com`). To store and process your data in the European Union only, use the URI `eu-vision.googleapis.com` in place of `vision.googleapis.com` for your REST API calls:

- `https://eu-vision.googleapis.com/v1/images:annotate`
- `https://eu-vision.googleapis.com/v1/images:asyncBatchAnnotate`
- `https://eu-vision.googleapis.com/v1/files:annotate`
- `https://eu-vision.googleapis.com/v1/files:asyncBatchAnnotate`

Setting the location using the client libraries

The Vision API client libraries accesses the global API endpoint (`vision.googleapis.com`) by default. To store and process your data in the European Union only, you need to explicitly set the endpoint (`eu-vision.googleapis.com`). The code samples below show how to configure this setting.

REST & CMD LINE | C# | GO | JAVA | NODE.JS | PHP | PYTH

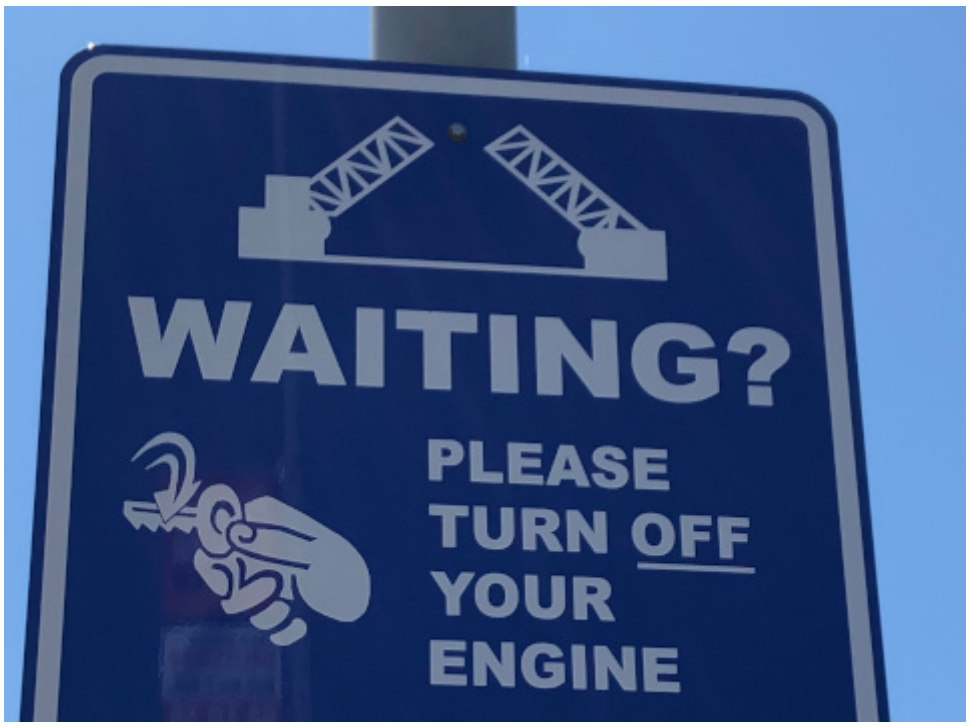
'PYTHON-DOCS-SAMPLES/BLOB/MASTER/VISION/CLOUD-CLIENT/DETECT/SET_ENDPOINT.PY)

```
from google.cloud import vision

client_options = {'api_endpoint': 'eu-vision.googleapis.com'}

client = vision.ImageAnnotatorClient(client_options=client_options)
```

Try it

Try text detection and document text detection below. You can use the image specified already (`gs://cloud-samples-data/vision/ocr/sign.jpg`) by clicking **Execute**, or you can specify your own image in its place.

To try document text detection, update the value of `type` to `DOCUMENT_TEXT_DETECTION`.



Try this API

Call this method on live data to see the API request and response. Need help with the
support page.

Request parameters

No method-level parameters

Show standard parameters

Request body

```
{
  "requests": [
    {
      "features": [
        {
          "type": "TEXT_DETECTION"
        }
      ],
      "image": {
        "source": {
          "imageUri": "gs://cloud-samples-data/vision/ocr/sign.jpg"
```

```
        }
      }
    }
  ]
}
```

For suggestions, press control+space or click one of the blue "add" circles.

Credentials

Google OAuth 2.0

OAuth 2.0 provides authenticated access to an API. Show scopes

API key

An API key is a unique string that lets you access an API.

EXECUTE

By clicking above, I agree that my use of the API Explorer is governed by the Terms and Privacy Policy.

上次更新：一月 10, 2020。