Spectrum Technology

V1.0 2019/12/25

抜粋版

はじめてのクラウドAI開発キット2

～グーグルのAIノウハウを最大活用、動画高速処理が可能に～

開発編

スペクトラム・テクノロジー株式会社

https://spectrum-tech.co.jp

sales@spectrum-tech.co.jp

all rights reserved 2019 spectrum technology co.

ST Spectrum Technology

開発編　目次

抜粋版のためページと内容は一致しません

all rights reserved 2019 spectrum technology co.

Spectrum Technology

開発編　目次

抜粋版のためページと内容は一致しません

all rights reserved 2019 spectrum technology co.

RaspberryPi運用マニュアル

1. Raspberry Piについて

   既に全世界で1000万台以上販売された手のひらサイズのコンピュータです。LinuxベースのRasbianOSで動作しております。

2. Linux基本コマンド

   ① システム関係

   - 起動：電源を入れると自動で起動します。
   - 再起動：$ reboot

     又は、menu>shutdown>reboot；左上のメニューから
   - 終了： $ shutdown

     又は、menu>shutdown>shutdown；左上のメニューから
   - ログアウト $ exit

     又は、menu>shutdown>logout；左上のメニューから
   - 日本語／英語の入力切替：キーボードのCTLとjを同時に押します（コントロール：左下とj）

all rights reserved 2019 spectrum technology co.

RaspberryPi運用マニュアル

2. Linux基本コマンド

② ディレクトリ操作、コピー、移動、削除

pi@raspberrypi:~$ cd /home/pi/Documents　ディレクトリの切り替え

pi@raspberrypi:/home/pi/Documents$ ls　ファイルとディレクトリの表示（表示したら操作したいファイルを右クリックでコピペして操作します

pi@raspberrypi:~$ cp ファイル名　ディレクトリ　配下のディレクトリのファイルを別のディレクトリへコピー

pi@raspberrypi:~$ mv ファイル名　ディレクトリ　配下のディレクトリのファイルを別のディレクトリへ移動

pi@raspberrypi:~$ sudo rm ファイル名　ファイルの削除

便利な機能　rm –help　コマンドのオプションが分からない場合は、ヘルプで問い合わせる。すべてのコマンド共通(マイナスを2個とhelp)

③ ユーザ権限、プロセス他

pi@raspberrypi:~ $ su -　スーパーユーザ（root)に切り替え、パスワードを入力

pi@raspberrypi:~$ ps a　現状の動いているプロセスを表示

pi@raspberrypi:~$ kill　特定のプロセスを強制終了

pi@raspberrypi:~$ sudo apt-get install pkg　パッケージのインストールなどに使用

pi@raspberrypi:~$ date　日付、時間の設定を行います。

pi@raspberrypi:~$ sudo leafpad /etc/network/interfaces　インタフェースに記述してい内容を変更します。Viよりも使いやすいです。

④ モジュール、usb、メモリ、HDDなどの表示

pi@raspberrypi:~$ lsmod　linuxのモジュールリスト表示

pi@raspberrypi:~$ lsusb　usbのデバイス表示

pi@raspberrypi:~$ free –mt　メモリ使用状態表示

pi@raspberrypi:~$ df　HDD（マイクロSD）の使用状態表示

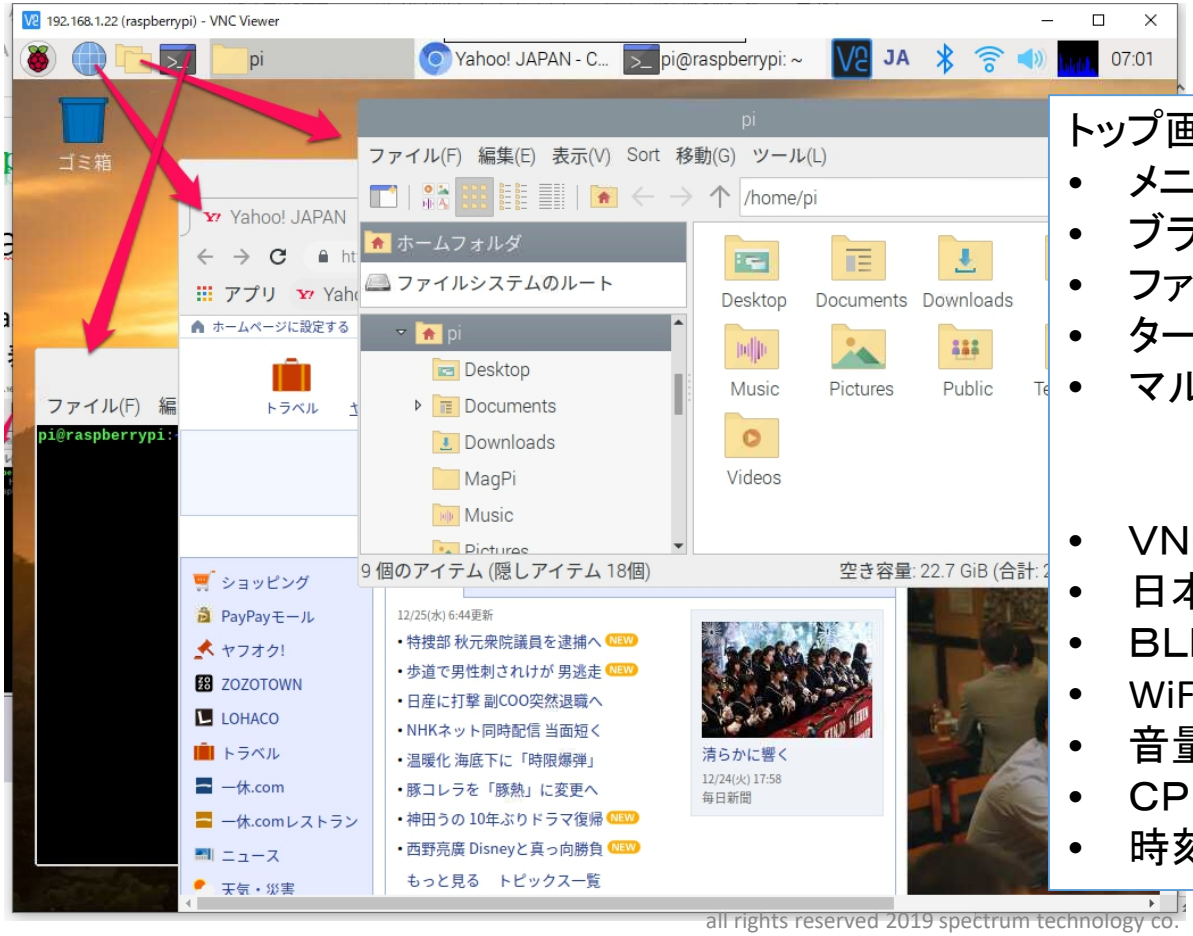
all rights reserved 2019 spectrum technology co.

RaspberryPi運用マニュアル

3. Raspberry Piの基本操作

① 表示画面と内容

デスクトップ上によく使うコマンド.txtがあります。
コピペして使ってください



トップ画面（上段のタスクバーで選択）

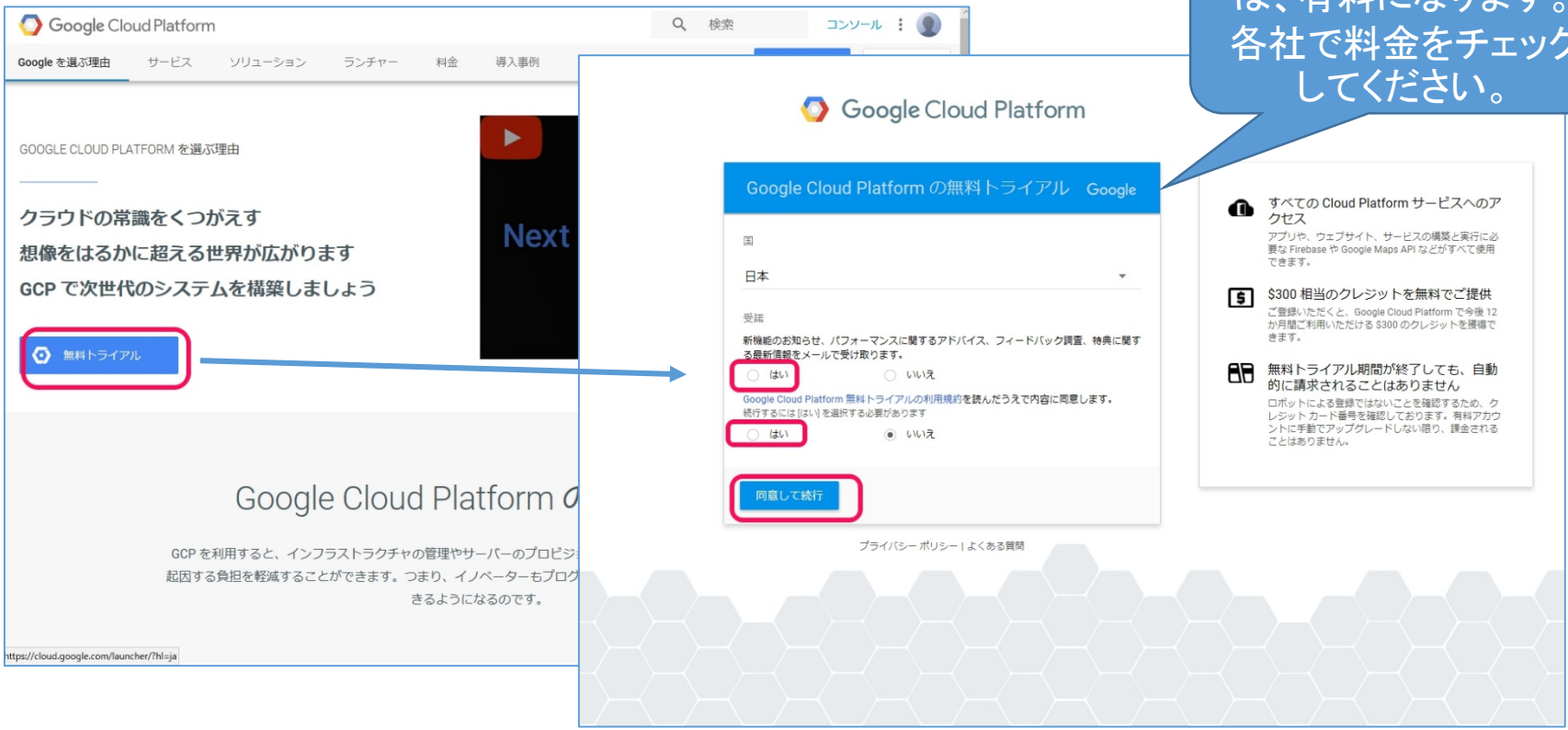
- メニュー
- ブラウザ
- ファイルマネジャ
- ターミナル
- マルチ画面選択

- VNC
- 日本語入力
- BLE
- WiFi
- 音量
- CPU使用率
- 時刻

all rights reserved 2019 spectrum technology co.

GCP設定（Google cloud platform）
1.アカウント登録

・GCPにアクセスして、無料トライアルを申しこみ、アカウント登録を実施します
https://console.cloud.google.com/



無料トライアル終了後は、有料になります。各社で料金をチェックしてください。

all rights reserved 2019 spectrum technology co.

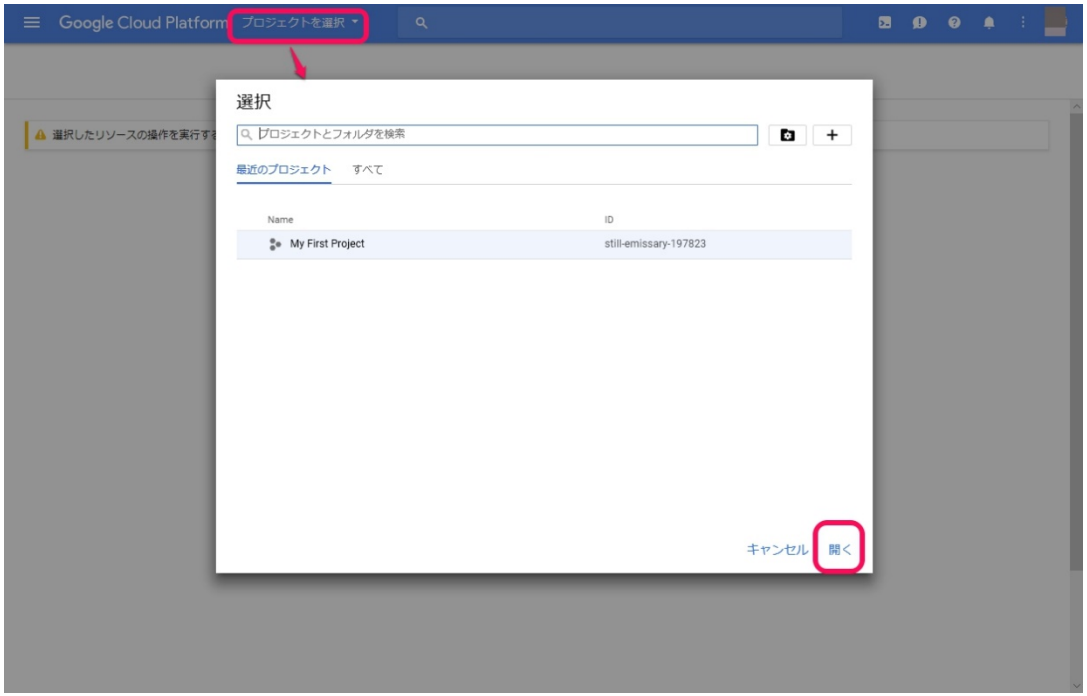
GCP設定（Google cloud platform）
1.アカウント登録

・住所、会社名、氏名、電話、クレジットカード情報を入力して、申し込みます

all rights reserved 2019 spectrum technology co.

GCP設定（Google cloud platform）
2.プロジェクト登録

・GCPのホームからプロジェクトを選択して、プロジェクト名を記入して、登録します。
・プロジェクト名とプロジェクトIDは作成後は、修正できません。

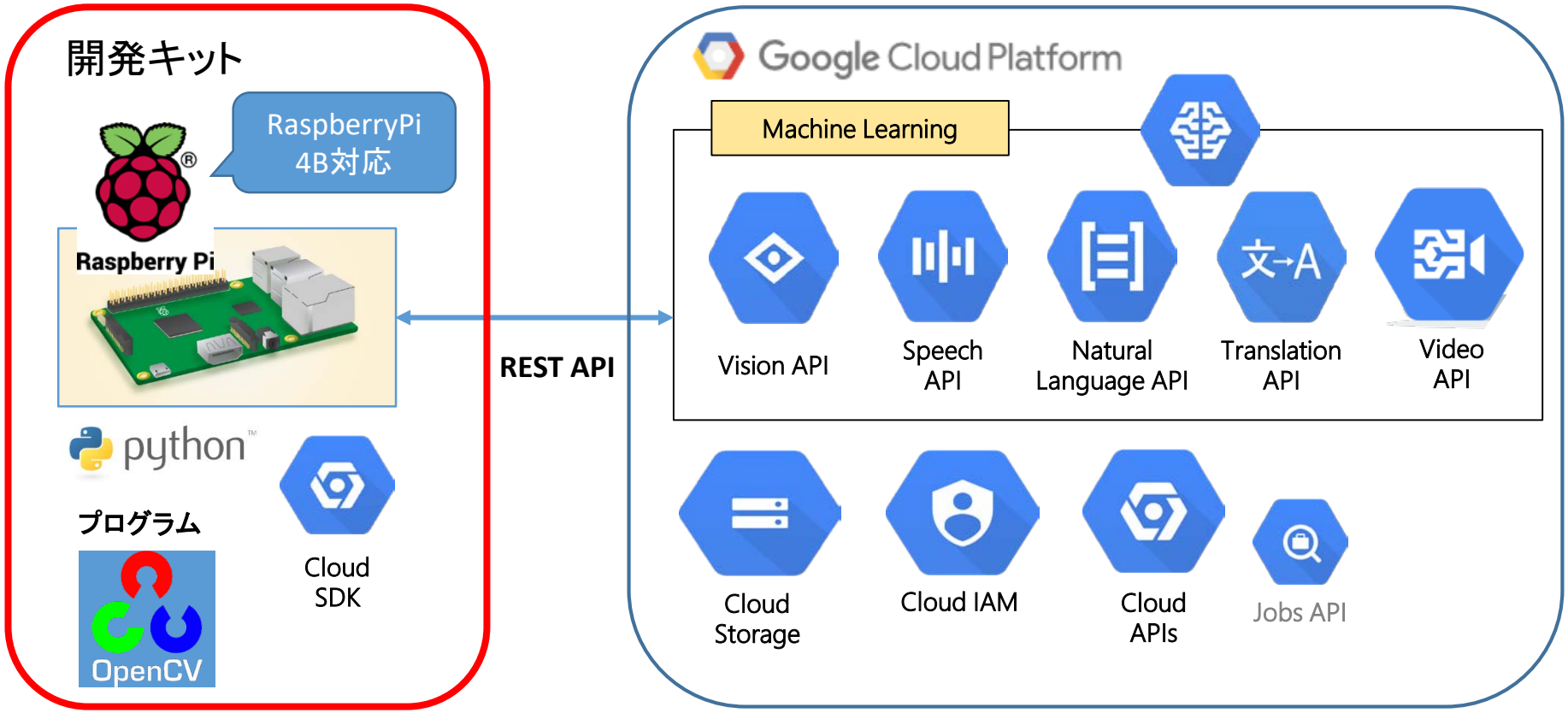


all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

1.開発キット全体像



all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

2. AI活用例

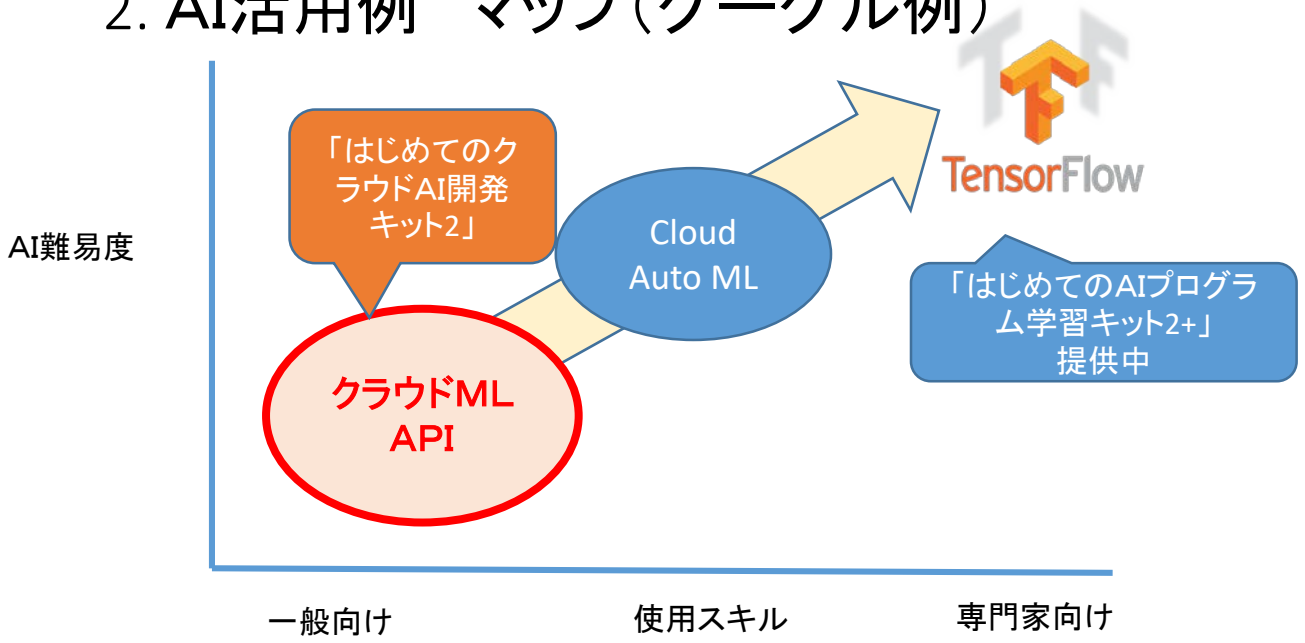
カテゴリ	活用例	要素技術	
画像認識	・OCRでの読み取り代替 ・写真のテキスト検出 ・顔検出による人数カウント ・不適切画像検出 ・類似デザイン検出 ・画像からWebサイト検出 ・画像へのインデックス付け	Ocr Ocr Face Safe Web label	
音声認識	・自動議事録作成 ・動画、音声ファイルからの文字起こし	Transcribe_stream Transcribe_async	
自然言語解析	・感情分析（音声認識との組合せ） ・エンティティ認識	Sentiment+Transbribe_async entity	
自動翻訳	・多言語翻訳 ・音声認識し、多言語で文字出力	Snippet Translate+transcribe_async	
動画認識	・不適切動画検出 ・ラベル検出 ・音声文字起こし	Adult Label Transcribe_async	

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

2. AI活用例　マップ（グーグル例）



サービス名	特徴	考慮事項
クラウドML　API	• グーグルのノウハウ（学習済モデル）を利用 • プログラム作成が簡単 • AI技術者が不要	・カスタマイズが難しい
Cloud Auto　ML	• クラウドAIとTensorflowの中間でカスタマイズが可能 • 高度なAI技術者が不要	
Tensorflow	• 自由にモデルを作成可能	• 学習データはユーザが準備 • AI技術者が必要

all rights reserved 2019 spectrum technology co.,ltd

Spectrum Technology

クラウドAI開発キット

3.開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

クラウドAI名	プログラム名	機能内容	実用度（当社評価）	信頼度（当社評価）	備考
Cloud vision api（画像認識）	OCR(文字認識）	画像内のテキストを検出、抽出できます。幅広い言語がサポートされており、言語の種類も自動で判別されます。	A	90%	手書き文字はC:10%位
	Face(顔検出）	画像に含まれる複数の人物の顔を検出できます。感情や帽子の着用といった主要な顔の属性についても識別されます。	A	90%	特定の人の顔認識はできません
	Label(ラベル検出）	乗り物や動物など、画像に写っているさまざまなカテゴリの物体を検出できます。	A	90%	
	Web(ウエブ検出）	類似の画像をインターネットで検索できます	A	90%	
	Safe(不適切画像検出）	アダルトコンテンツや暴力的コンテンツなど、画像に含まれる不適切なコンテンツを検出できます	A	90%	
	Logo(ロゴ検出）	画像に含まれる一般的な商品ロゴを検出できます	C	10%	文字のないロゴは検出できない
	Landmark（ランドマーク検出）	画像に含まれる一般的な自然のランドマークや人工建造物を検出できます。	B	50%(画像次第）	
	Prop(画像属性検出）	画像のドミナントカラーや切り抜きのヒントなど、画像の一般的な属性を検出できます	B		
	Crop(画像切り抜き）	認識できる画像に切り抜きできます。画像の最小化	B		
	multipage	複数画像を処理できるプログラム			
	loadjson	Json形式のデータを読み出すプログラム			

http方式とライブラリ方式の二つ準備

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット
3.開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

クラウドAI名	部品名	機能内容	実用度（主観）	信頼度（主観）	備考
Cloud speech api（音声認識、多言語対応）	Transcribe(同期音声認識）	ローカルに保存されている、短い音声（1 分未満）を処理し、認識されたテキストを出力します。	A	70%	固有名詞が難しい
	Transcribe_async(非同期音声認識）	Google Cloud Storage に保存されている、1 分より長い音声を認識し、テキストで出力します。 動画から音声テキストの検出も可能	A	70%	
	Transcribe_stream（ストリーミング音声認識）	ローカルのリアルタイムでストリーム音声認識します			ローカルなので上記の部品で代用可能
	Transcribe_stream_mic（マイク入力ストリーミング音声認識）	マイク入力で、リアルタイムで音声認識を行い、テキストを出力します。びっくりする能力。是非お試しください。 Googleドキュメントの音声入力と同じです。(*1)	A	70%	すごいです。
	pyaudio	マイクの試験用プログラム			

(*1)USBマイクは付いておりませんので、お客様で準備してください

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

3.開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

クラウドAI名	部品名	機能内容	実用度（主観）	信頼度（主観）	備考
Cloud Natural language api（自然言語処理、多言語）	Sentiment（感情分析）	テキストのブロック内で示されている全体的な感情を読み取ることができます。	B		
	Syntax（構文解析）	トークンと文の抽出、品詞（PoS）の特定、各文の係り受け解析ツリーの作成が可能です。	B		文字ずれ発生中：原因不明
	Entity(エンティティ認識）	エンティティとラベル（人、組織、場所、イベント、商品、メディアなど）を特定できます	A	90%	
	Classify(カテゴリ分類）英語のみ	事前定義された 700 以上のカテゴリでドキュメントを分類できます	A	90%	英語のみ
Cloud translate api（翻訳、多言語）	Quickstart（テキスト翻訳）	グーグルの自動翻訳。多言語対応。プログラム上で言語を指定	A	70%	
	Snippet(テキスト翻訳、多言語）	グーグルの自動翻訳。多言語を翻訳時に指定。	A	70%	

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

3.開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

クラウドAI名	部品名	機能内容	実用度（主観）	信頼度（主観）	備考
Cloud video api beta（動画認識）	Label（ラベル認識）	「犬」、「花」、「車」などの動画内のエンティティを検出します	A	70%	
	adult(不適切動画検出）	アダルト コンテンツや暴力的コンテンツなど、画像に含まれる不適切なコンテンツを検出できます	A	70%	
	analyze（統合型分析）	Shot:動画内のシーンの変更を検出します Adult:不適切シーンの検出	B		
	Face(顔認識）	プログラムはありますが、エラー。プライバシー保護のため	C		エラー
	音声認識	Speech>transcribe_asyncにて、動画の音声のみ取り出して、文字起こしが可能	A	70%	便利

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

3.開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

クラウドAI名	プログラム名	機能内容	実用度（当社評価）	信頼度（当社評価）	備考
Cloud video api（動画認識）	Label（ラベル認識）	「犬」、「花」、「車」などの動画内のエンティティを検出します	A	90%	
	adult(不適切動画検出）	アダルト コンテンツや暴力的コンテンツなど、画像に含まれる不適切なコンテンツを検出できます	A	90%	
	analyze（統合型分析）	Shot:動画内のシーンの変更を検出します Adult:不適切シーンの検出	B		
	音声認識	Speech>transcribe_asyncにて、動画の音声のみ取り出して、文字起こしが可能	A	80%	便利
	カメラ（リアルタイム）	リアルタイムで物体、顔のパーツを識別、高速で処理	A	90%	RaspberryPi4Bにより高速処理が可能

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

3.開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

区分名	プログラム名	機能内容	実用度（当社評価）	信頼度（当社評価）	備考
opencv	FaceDection（顔認識）	顔の目、顔全体を認識します。	A	90%	
	FaceRecognition(顔検出）	顔登録：4名まで顔を登録できます。20枚／人の顔を1分で登録します	A	90%	
		顔学習	A	90%	
		顔検出：登録した顔を検出し、確率を表示します。	A	80%	

ローカルで動作します。

all rights reserved 2019 spectrum technology co.

Spectrum Technology

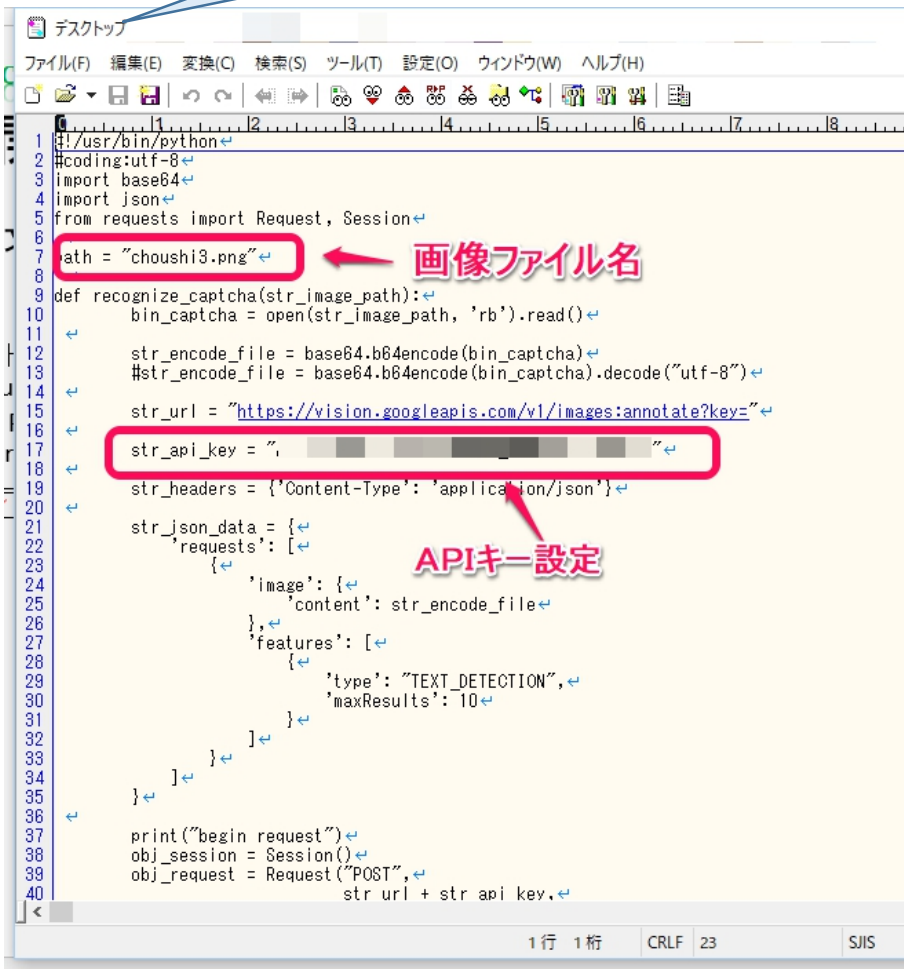
クラウドAI開発キット

4.使用方法

① Cloud vision>1_OCR(文字認識)

- テキスト認識：ocr1.py（画像からテキスト抽出）
  - APIキーの設定
  - Windows PC＞ネットワーク＞RaspberryPi＞Pi>Documents>cloud vision>1_ocrをクリック
  - Ocr1.pyをwindows PCのローカルにコピーして、さくらエディタなどでocr1.pyを開きます。
  - 17行のstr_api_key=“ xxxxx”にGCP設定>4.APIキー設定で取得したキー（15ページ）を貼り付けます
  - また、画像ファイル名は、7行目です。適宜書き換えて使用してください。
  - また、データ出力は、53行目になります。デフォルトは、data1.jsonです。
  - 修正が完了したら、再度raspberry Piにocr1.pyを上書きしてください。

さくらエディタ
https://sourceforge.net/projects/sakura-editor/



all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

4.使用方法

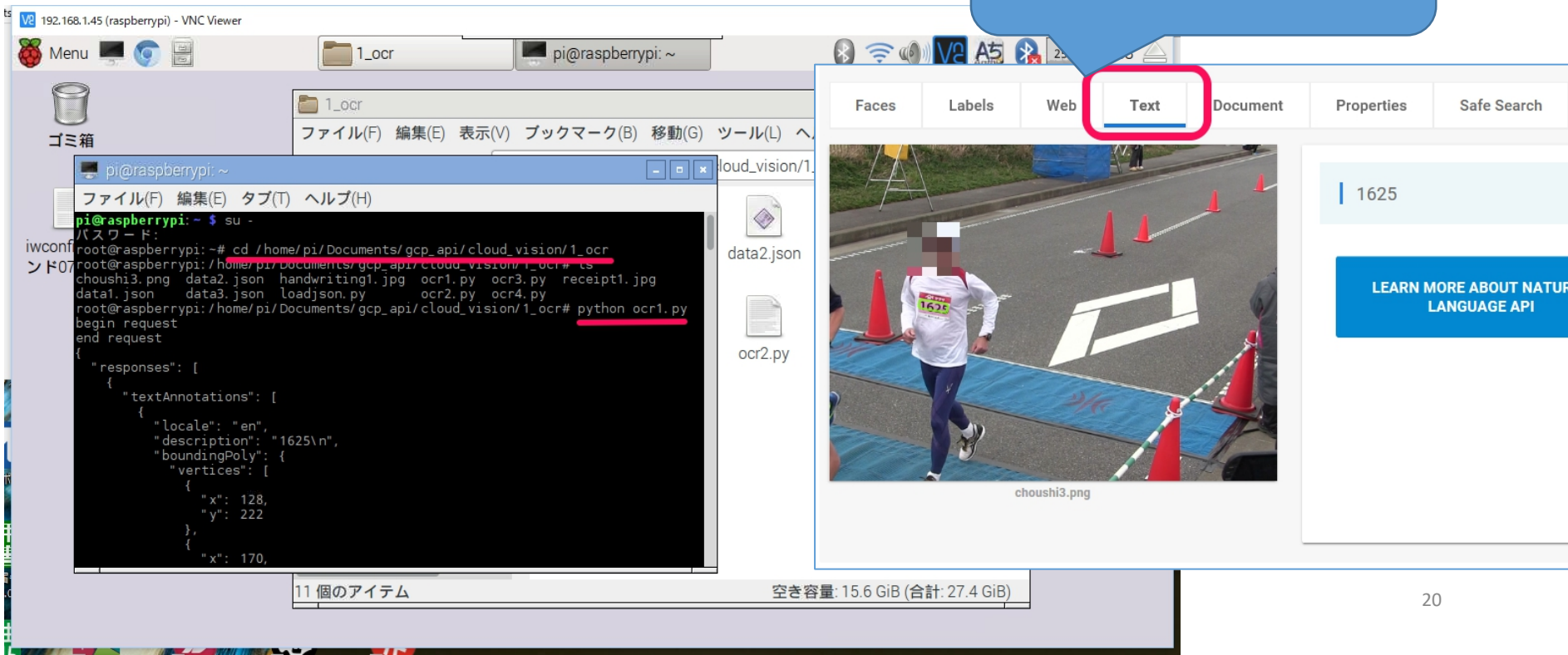
pyrhon3で実施してください

① Cloud vision>1_OCR(文字認識)

- テキスト認識：ocr1.py（画像からテキスト抽出）
  - Raspberry piからocr1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata1.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/1_ocr
$ python3 ocr1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja



Spectrum Technology

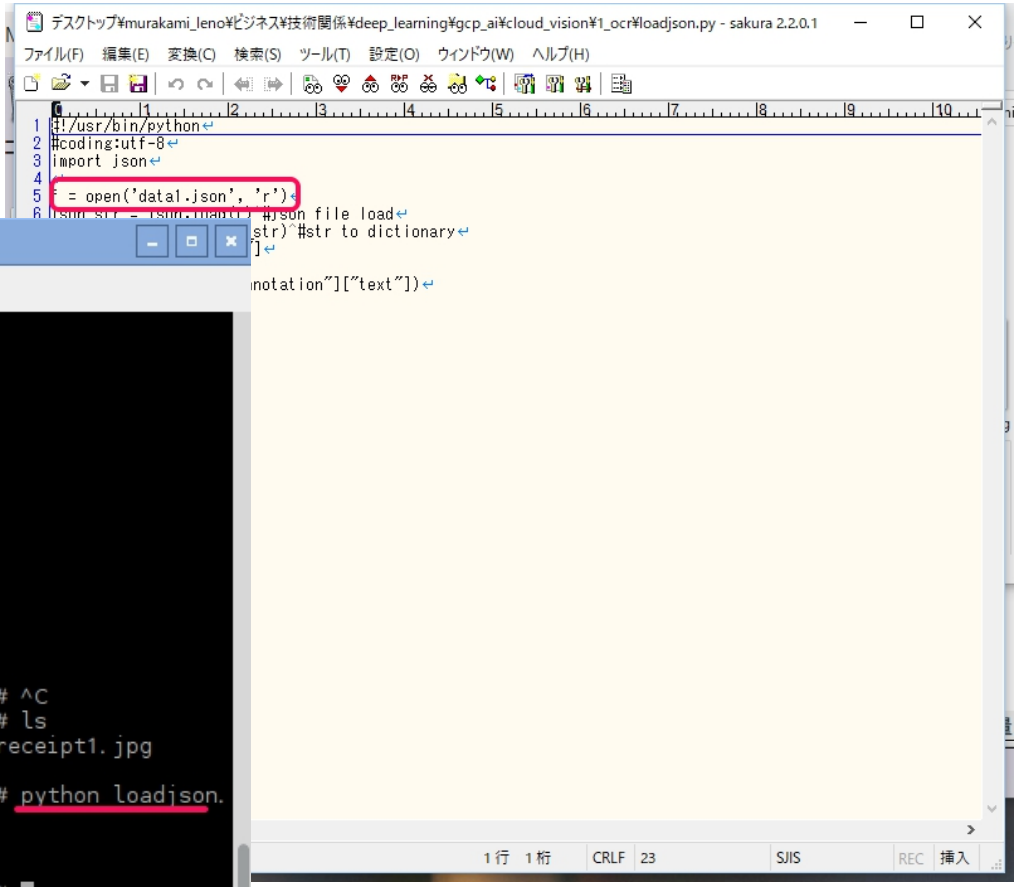
クラウドAI開発キット

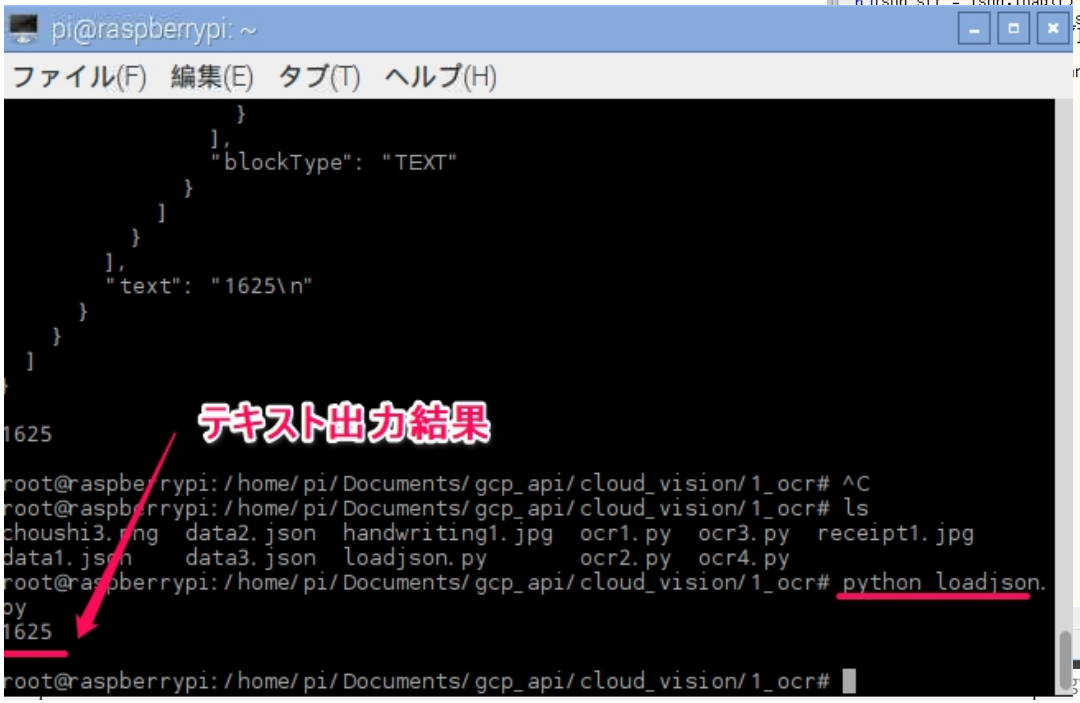
4.使用方法

① Cloud vision>1_OCR(文字認識)

- テキスト認識：ocr1.py（画像からテキスト抽出）
  - jsonデータを、loadjson.pyでテキストのみ取り出します。
  - 読み出すデータは5行目に定義。
  - 必要に応じて書き換えてください。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/1_ocr
$ python3 loadjson.py





Spectrum Technology

クラウドAI開発キット

4.使用方法

① Cloud vision>1_OCR(文字認識)

- テキスト認識：ocr2.py（文字読み取り）
  - 同様にApi キーを設定します。
  - Raspberry piからocr2.pyを実行します。
  - 出力は、piの画面上とファイルのdata2.jsonの双方にでます。
  - loadjson.pyで読み出しできます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/1_ocr
$ python3 ocr2.py
$ python3 loadjson.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

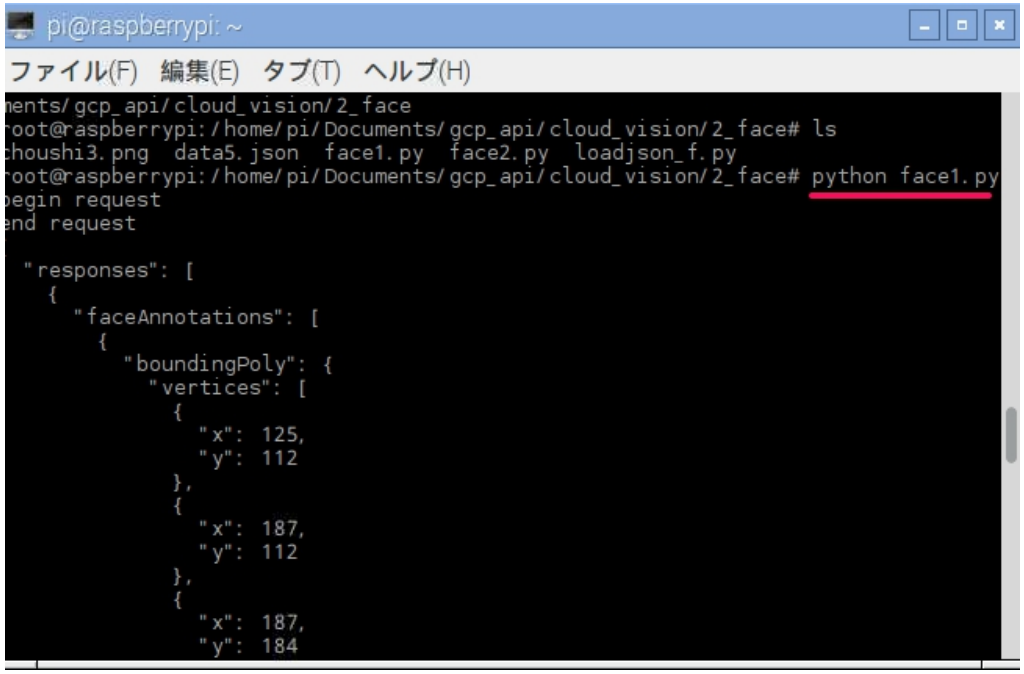
4.使用方法

① Cloud vision>2_face(顔検出）

- face1.py（http使用）
  - Raspberry piからface1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata5.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/2_face
$ python3 face1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja



顔認識は、できません。プライバシー保護

all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

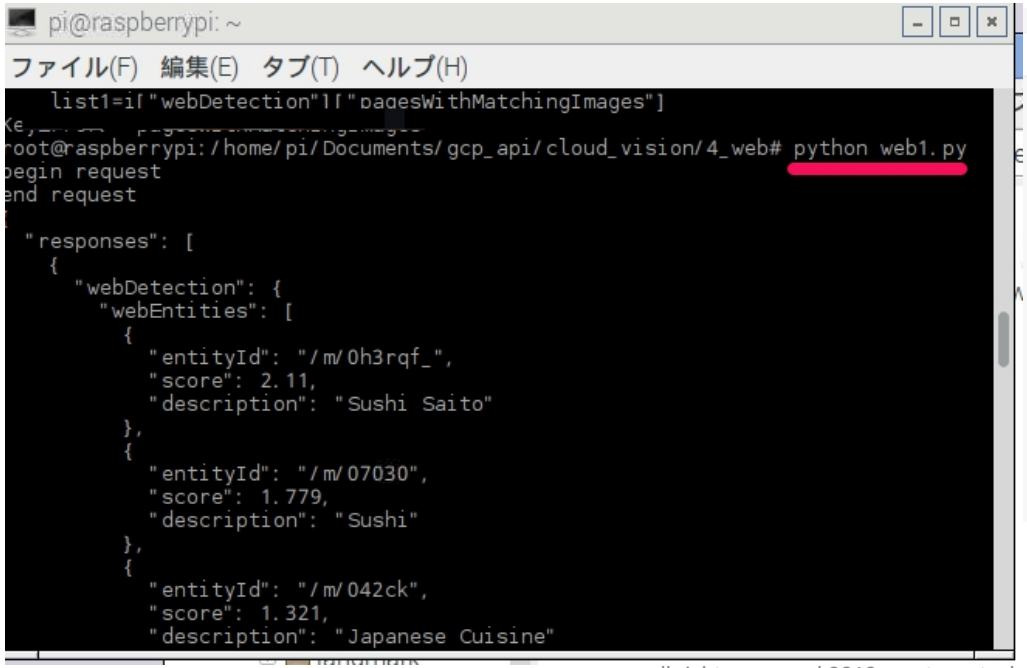
4.使用方法

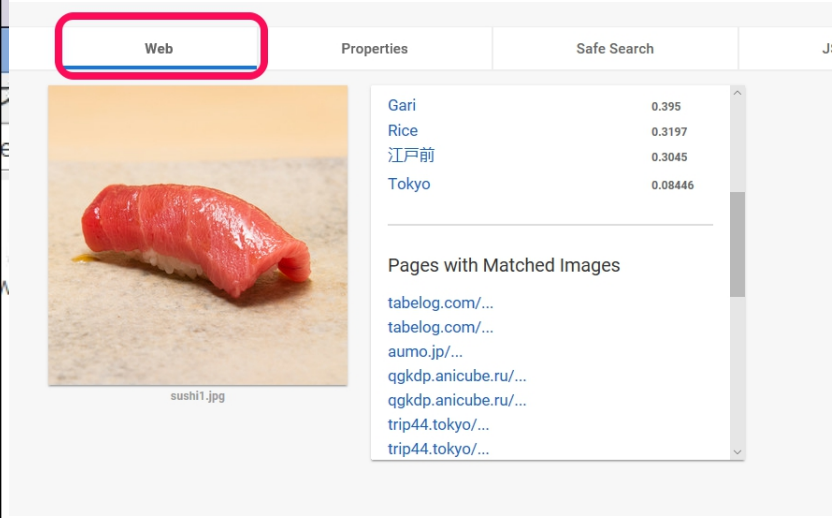
① Cloud vision>4_web（web検出）

- web1.py（http使用）
  - Raspberry piからweb1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata7.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/4_web
$ python3 web1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





Web上で類似した画像、URLが出力されます

all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

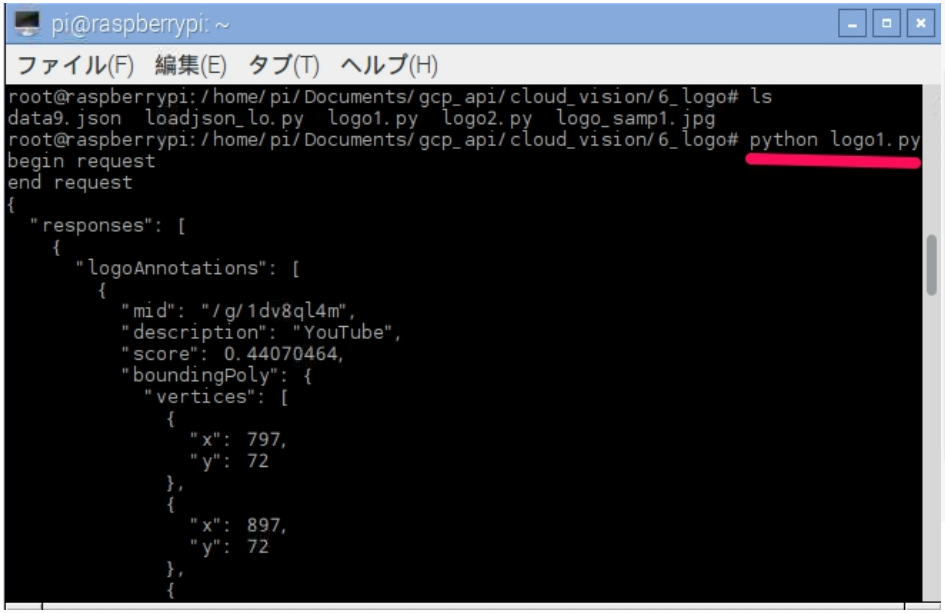
4.使用方法

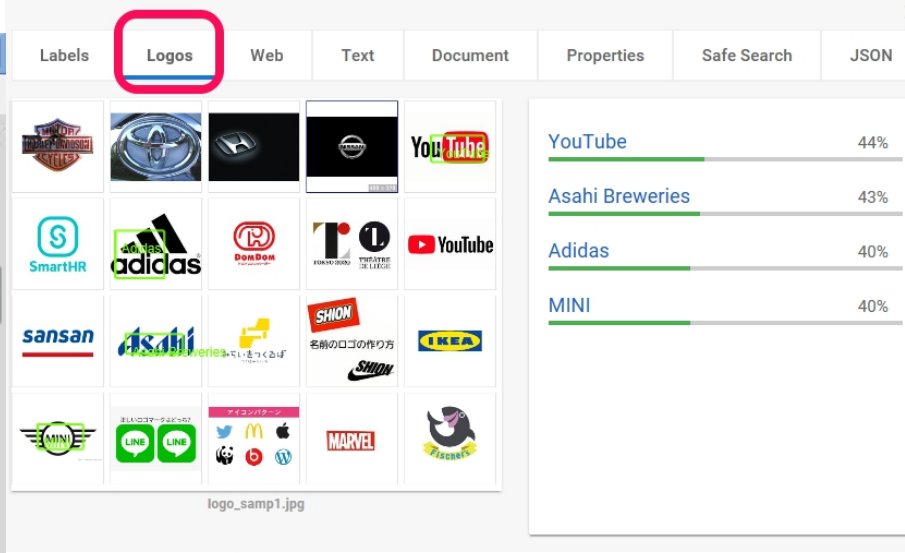
① Cloud vision>6_logo（ロゴ検出）

- logo1.py（http使用）
  - Raspberry piからlogo1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata9.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/6_logo
$ python3 logo1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





ロゴ検出できます。テキストがない場合は検出できません

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

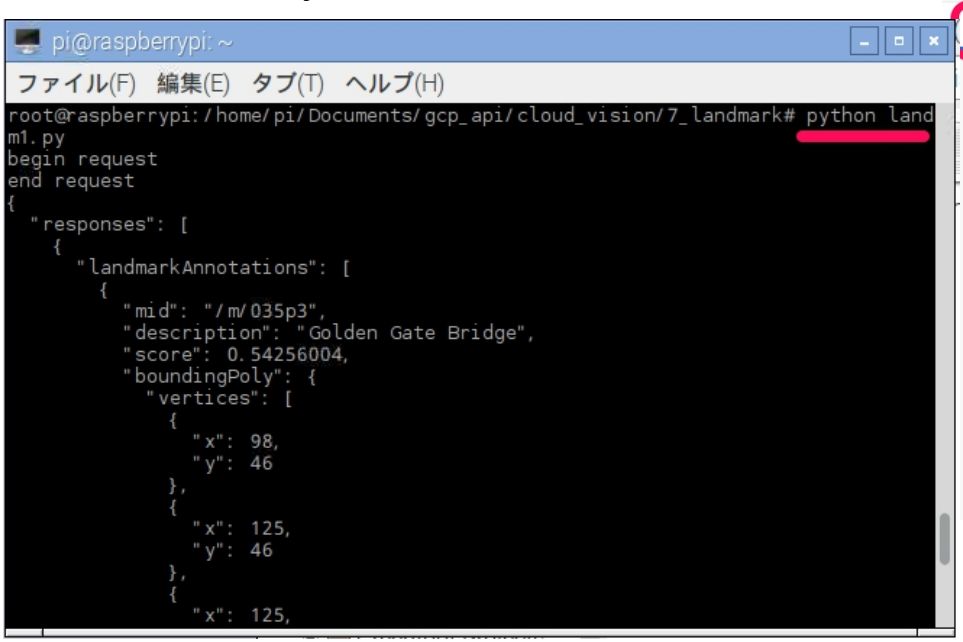
4.使用方法

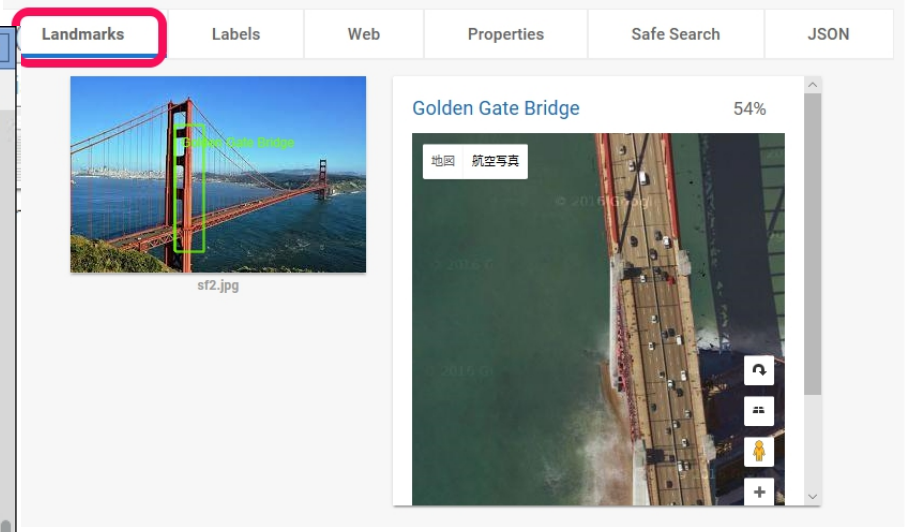
① Cloud vision>7_landmark（ランドマーク検出）

- landm1.py（http使用）
  - Raspberry piからlandm1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata11.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/7_landmark
$ python3 landm1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





ランドマークが検出できます。

all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

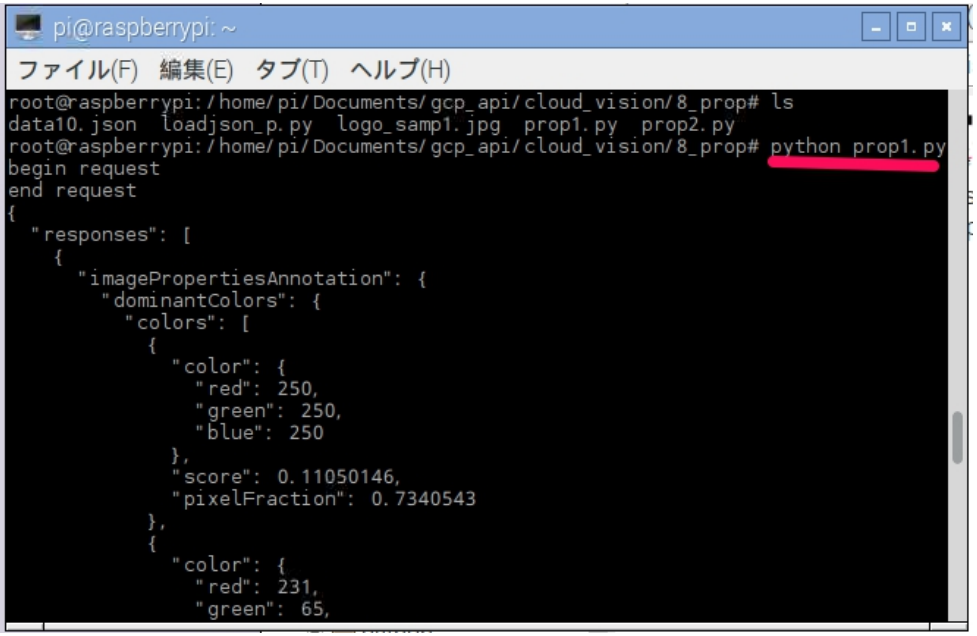
4.使用方法

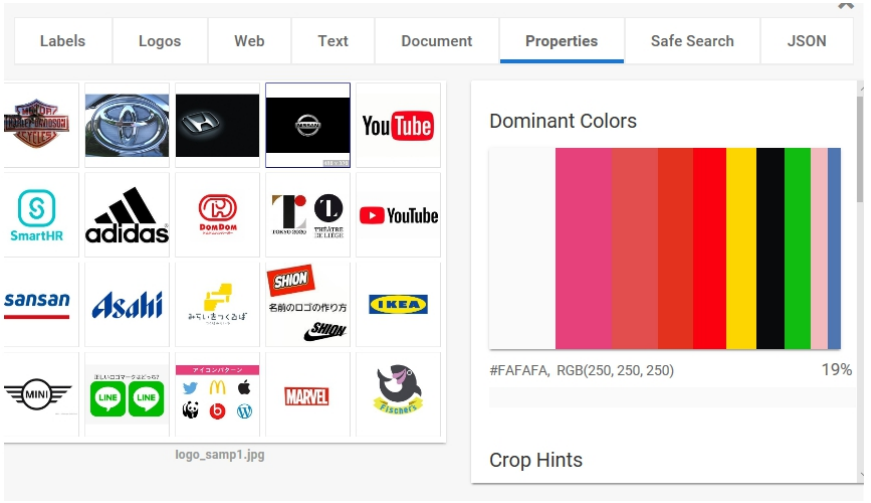
① Cloud vision>8_prop（プロパティ検出）

- Prop1.py（http使用）
  - Raspberry piからprop1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata10.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/8_prop
$ python3 prop1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





画像の色などのプロパティ情報が検出できます。

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

4.使用方法

① Cloud vision>9_crop（切り出し検出）

- Crop2.py（ライブラリ使用）
  - Raspberry piからpython3 crop2.py choushi3.jpg cropを実行します。
  - 画像ファイルは、crop2.pyの後に設定します。Jpgのみ有効です。
  - 出力は、output-crop.jpgです。

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/9_crop
$ python3 crop2.py choushi3.jpg crop

choushi3.jpg

output-crop.jpg

all rights reserved 2019 spectrum technology co.

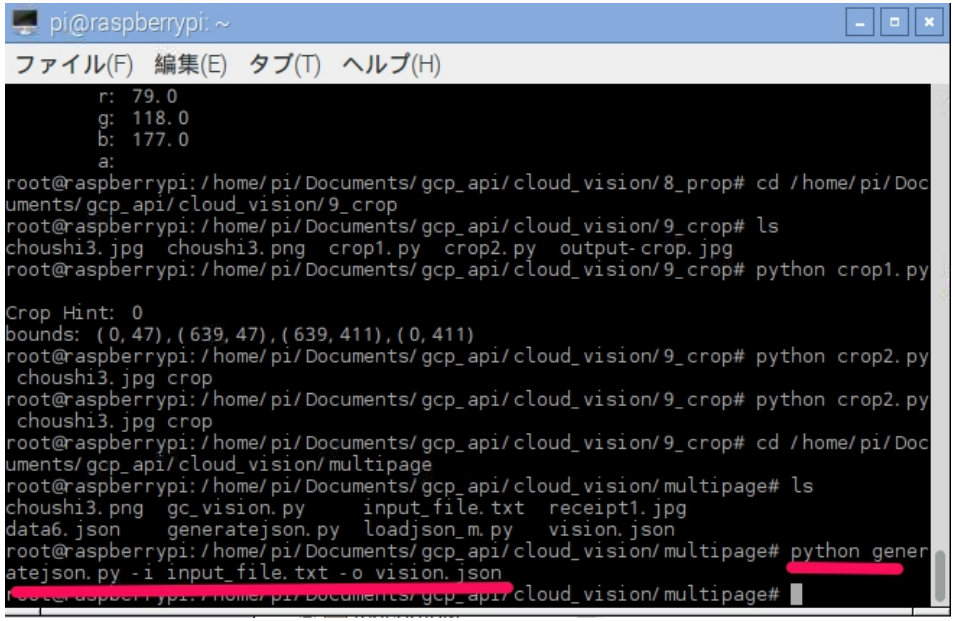
クラウドAI開発キット

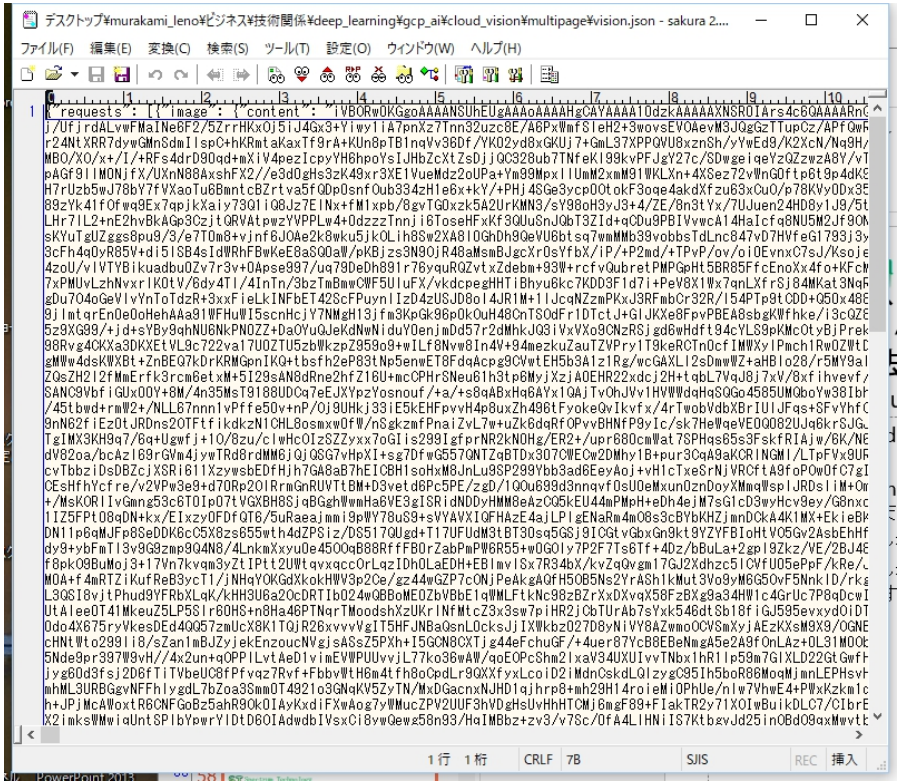
4.使用方法

① Cloud vision>multipage（複数画像同時処理）

- Vision.json（cloud visionに送信するファイル作成）
  - Generatejson.pyを使用して、送信ファイルを作成します。
  - 入力ファイル名は、input_file.txt
  - 出力ファイル名は、vision.json（任意に変えられます）

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_vision/multipage
$ python3 generatejson.py -i input_file.txt -o vision.json





vision.json

all rights reserved 2019 spectrum technology co.

Spectrum Technology

クラウドAI開発キット

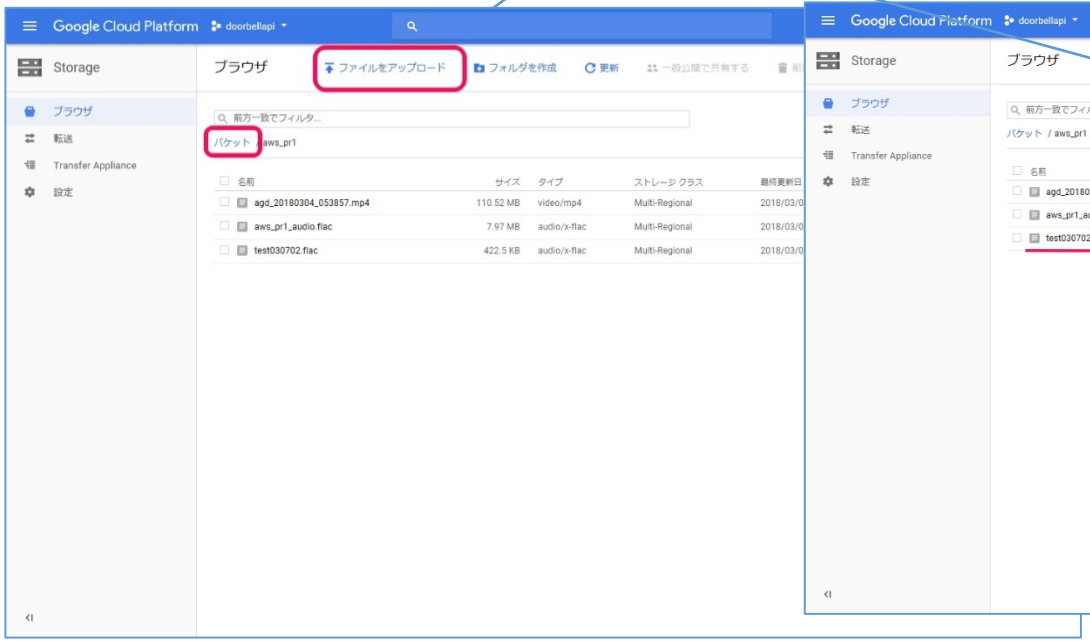
4.使用方法

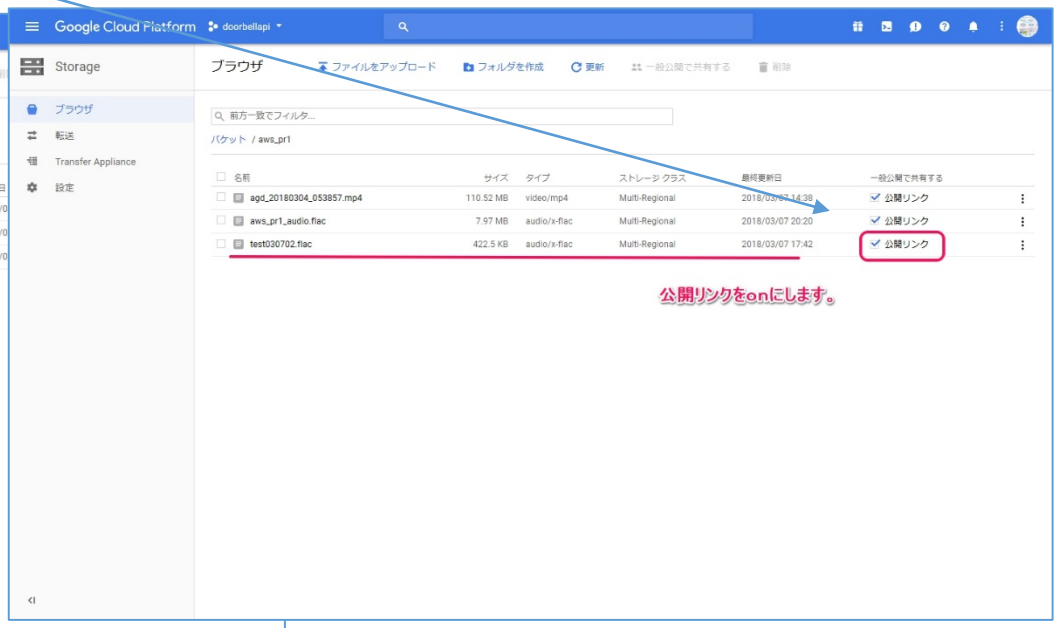
② Cloud speech>transcribe（短時間音声認識）

- transcribe.py（google storage）
  - flacなどの音声ファイルをグーグルのストレージに保存して、で日本語テキストに変換
  - バケット＞ファイルをアップロード
  - 公開リンクをONへ
  - リンク名は、gs://バケット名/test030702.flac

コマンド
$ cd
/home/pi/Documents/gcp_api/cloud_speech/
$ python3 transcribe.py gs://バケット名
/test030702.flac

音声ファイル変換サイト
https://online-audio-converter.com/ja/





all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

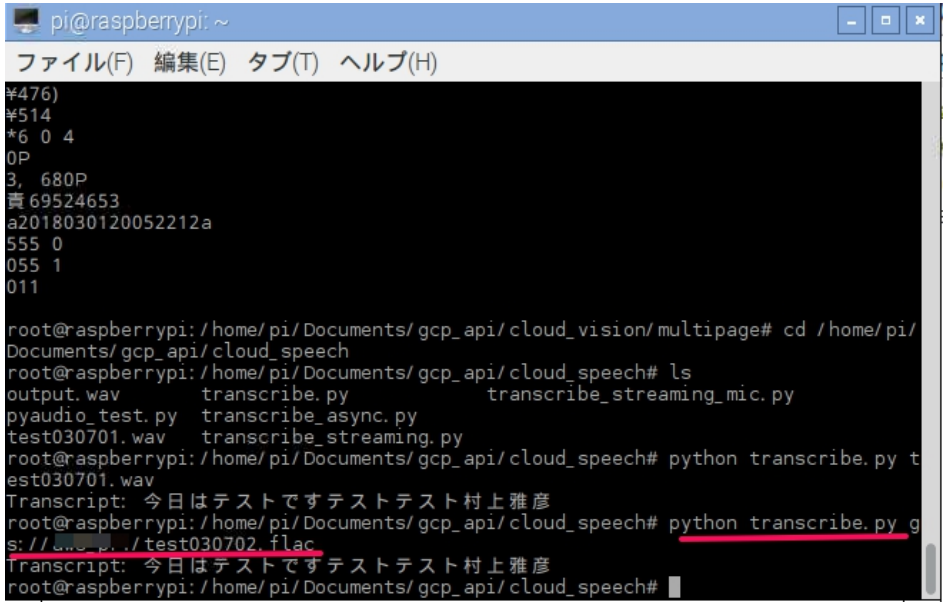
4.使用方法

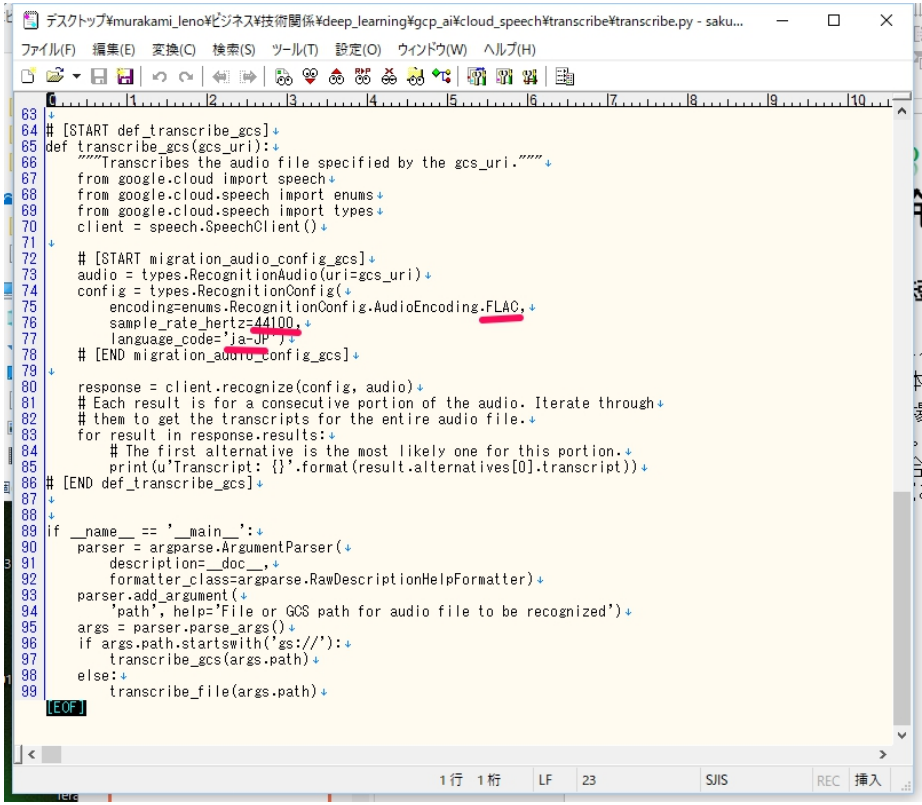
② Cloud speech>transcribe（短時間音声認識）

- transcribe.py（google storage）
  - flacなどの音声ファイルをグーグルのストレージに保存して、で日本語テキストに変換
  - 音声ファイルを作成する場合は、1ch, 44.1KHzで行ってください。
  - コーデックを変更する場合は、76行の値を変更してください。Flacのみになります

コマンド
$ cd /home/pi/Documents/gcp_api/cloud_speech/
$ python3 transcribe.py gs://バケット名/test030702.flac

音声ファイル変換サイト
https://online-audio-converter.com/ja/





all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

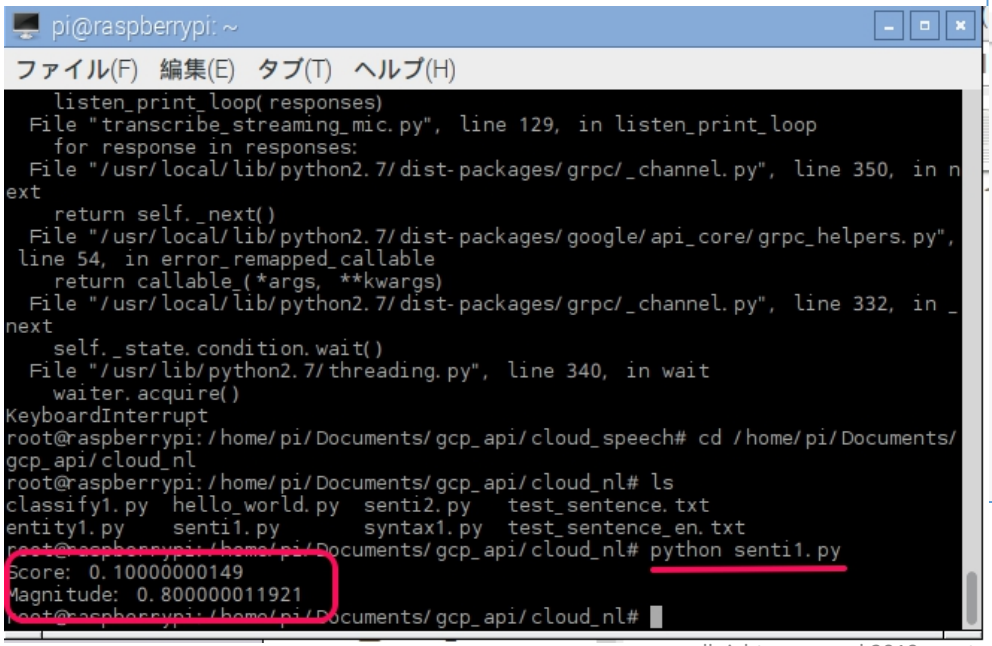
4.使用方法

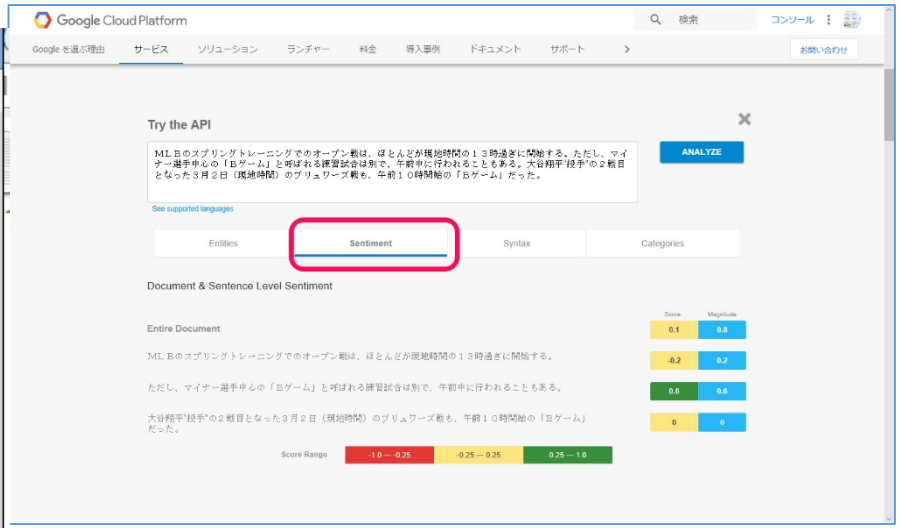
③ Cloud nl>sentiment（感情分析）

- senti1.py（全文解析）
  - sentiment1.pyを実行します。
  - 14行目にテキストのファイル名を設定しています。test_sentence.txtを添付しています。
  - スコア（ポジティブ、ニュートラル、ネガティブ）と感情の強度がでます。詳細は以下のURL参照。
  - https://cloud.google.com/natural-language/docs/basics?authuser=1&hl=ja#interpreting_sentiment_analysis_values

コマンド
$ cd
/home/pi/Documents/gcp_api/cloud_nl/
$ python3 senti1.py

Webでの確認
https://cloud.google.com/natural-language/?authuser=1&hl=ja





all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

4.使用方法

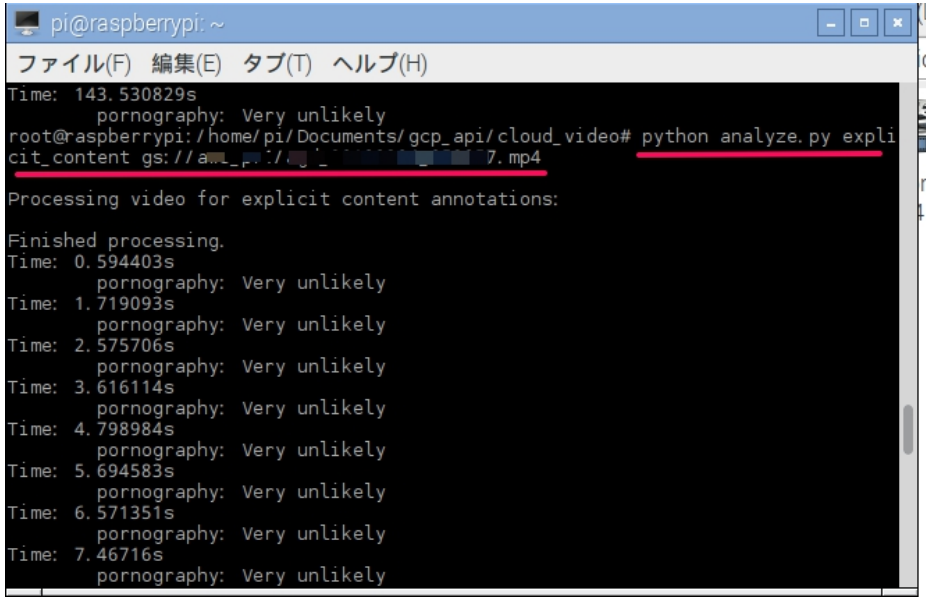
⑤ Cloud video

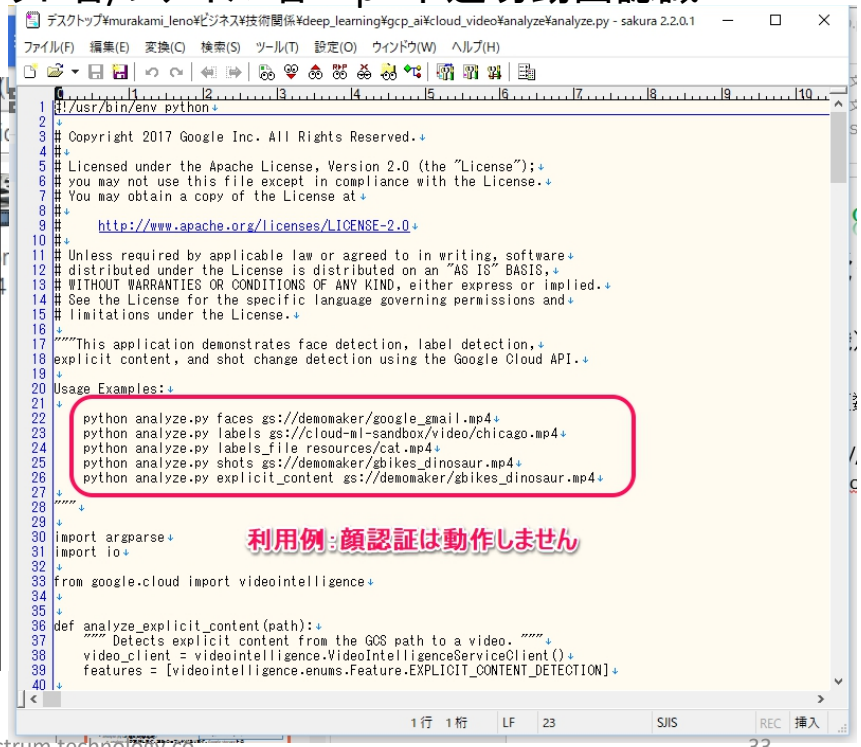
- analyze.py（複数動画認識）
  - analyze.pyを実行します。複数のコマンドがあります。Google storage上のファイルで動作します。

$ python3 analyze.py labels gs://バケット名/ファイル名.mp4　ラベル認識

$ python3 analyze.py explicit_content gs://バケット名/ファイル名.mp4不適切動画認識

コマンド
$ cd
/home/pi/Documents/gcp_api/cloud_video/
$ python3 analyze.py explicit_content gs://
バケット名/ファイル名.mp4





all rights reserved 2019 spectrum technology co.

クラウドAI開発キット

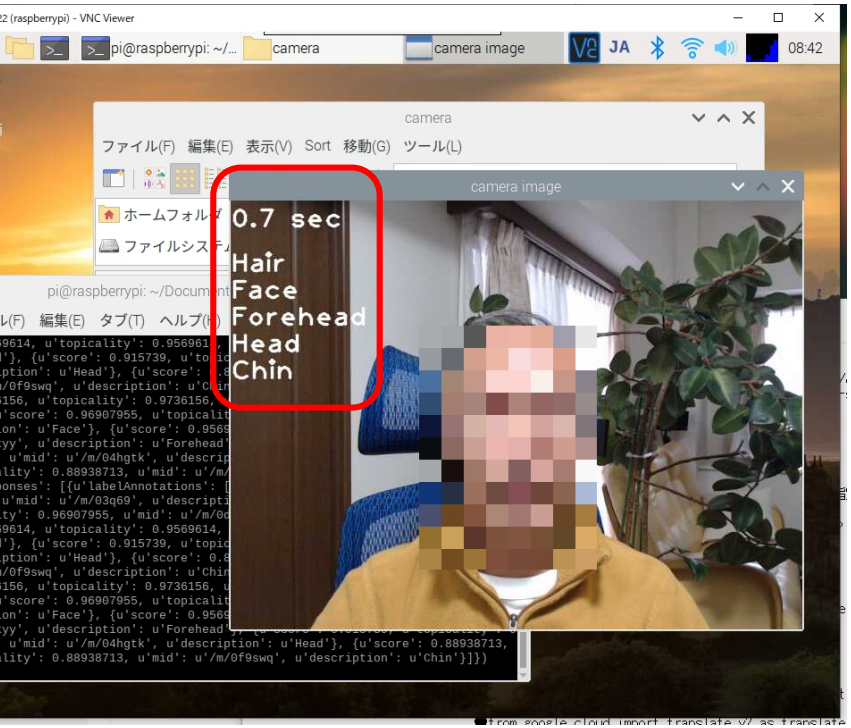
4.使用方法

⑤ Cloud video

- object_cam1.py（リアルタイム・カメラによる物体認識）
  - Webカメラで映ったラベルを検出します。
  - 16行目にAPIキー
  - RaspberryPi4Bで高速処理が可能

コマンド
$ cd
/home/pi/Documents/gcp_api/cloud_video/
$ python object_cam1.py

python3でエラーのためpythonで実施してください



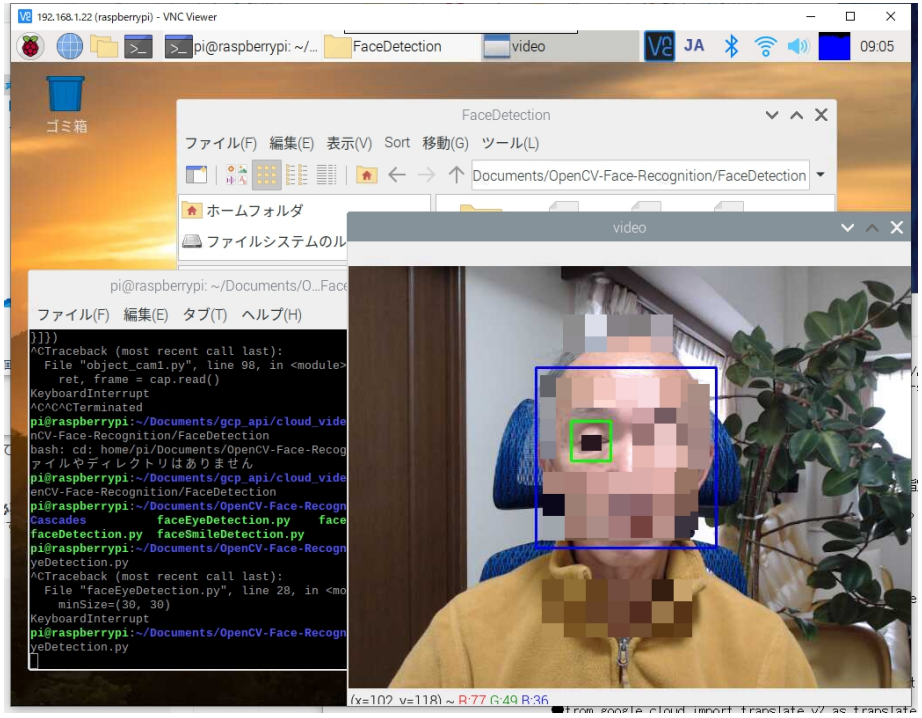

```
#! /usr/bin/python
# -*- coding: utf-8 -*-

# Google Cloud Vision API：LABEL_DETECTION

import sys
import base64
import cv2

from requests import Request, Session
import json
import time
import threading

api_key = 'AIzaSyDpkR                    p'

# 検出する顔の数の最大数（増やすとレスポンスが遅くなる）
max_results = 5

# END POINTS
DISCOVERY_URL = 'https://vision.googleapis.com/v1/images:annotate?key='

# cv画像と画像ファイルへのPathと検出最大数が引数
def googleAPI(img, max_results):
    # 通信不良等を考慮してTry/expectしておく
    try:
        # カメラ画像をJPG画像へ変換
        retval, image = cv2.imencode('.jpg', img)

        # Headerやpayload
        str_headers = {'Content-Type': 'application/json'}
        batch_request = {'requests': [{'image': {'content': base64.b64encode(image)}, 'features': [{'type': 'LABEL_DETECTI

        # セッション作ってリクエストSend
        obj_session = Session()
        obj_request = Request("POST", DISCOVERY_URL + api_key, data=json.dumps(batch_request), headers=str_headers)
        obj_prepped = obj_session.prepare_request(obj_request)
        obj_response = obj_session.send(obj_prepped, verify=True, timeout=180)

        # Responseからjsonを抽出
        response_json = json.loads(obj_response.text)
```

all rights reserved 2019 spectrum technology co.

Opencv開発
1. Opencv
①顔認識（カメラ）

コマンド
$ cd /home/pi/Documents/OpenCV-Face-Recognition/FaceDetection

- Opencvを使ったwebカメラでの顔認識です。
- https://www.instructables.com/id/Real-time-Face-Recognition-an-End-to-end-Project/
- python3 faceDetection.py
- python3 faceEyedetection.py



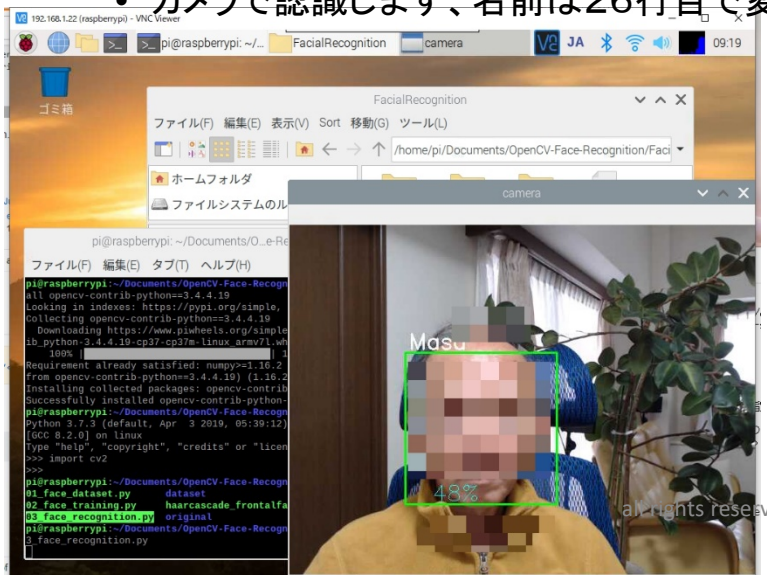
all rights reserved 2019 spectrum technology co.

Opencv開発

1. Opencv
②顔検出（カメラ）

コマンド
$ cd /home/pi/Documents/OpenCV-Face-Recognition/FacialRecognition

- Opencvを使った、webカメラでの登録した顔を検出します。
- https://www.instructables.com/id/Real-time-Face-Recognition-an-End-to-end-Project/
- 顔の登録
- python3 01_face_dataset.py
  - Face id：0-3までを入力してください。
  - 自動で顔のスキャンが始まります。数分で終了します。
- python3 02_face_training.py
- python3 03_face_recognition.py
  - カメラで認識します、名前は26行目で変更可能




```
Real Time Face Recogition
    ==> Each face stored on dataset/ dir, should have a unique numeric integer ID as 1, 2, 3, etc
    ==> LBPH computed model (trained faces) should be on trainer/ dir
Based on original code by Anirban Kar: https://github.com/thecodacus/Face-Recognition

Developed by Marcelo Rovai - MJRoBot.org @ 21Feb18

'''

import cv2
import numpy as np
import os

recognizer = cv2.face.LBPHFaceRecognizer_create()
recognizer.read('trainer/trainer.yml')
cascadePath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascadePath);

font = cv2.FONT_HERSHEY_SIMPLEX

#iniciate id counter
id = 0

# names related to ids: example ==> Marcelo: id=1,  etc
names = ['None', 'Masa', 'Paula', 'Ilza', 'Z', 'W'] 

# Initialize and start realtime video capture
cam = cv2.VideoCapture(0)
cam.set(3, 640) # set video widht
cam.set(4, 480) # set video height

# Define min window size to be recognized as a face
minW = 0.1*cam.get(3)
minH = 0.1*cam.get(4)

while True:

    ret, img =cam.read()
    #img = cv2.flip(img, -1) # Flip vertically

    gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

    faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor = 1.2
```

all rights reserved 2019 spectrum technology co.