IBM Phase-1 Project

Image recognition with IBM cloud visual Recognition:

**What is image recognition?**

Image recognition, in the context of machine vision, is the ability of software to identify objects, places, people, writing and actions in digital images. Computers can use machine vision technologies in combination with a camera and artificial intelligence ([AI](https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence)) software to achieve image recognition…

Many times, you want to search a restaurant which serves a specific dish. And you know the possibility of remembering name of that dish is bleak, but you took a picture last time you had it. In such situation a feature to search a dish by image is sorely missed. In this blog we are going to learn how to create an application to find restaurants with the help of food image you have. You will also learn how to work with IBM Watson for Visual Recognition and use HERE location services for maps and location analytics.

## Lab overview:

## Scenario:

IBM Watson Visual Recognition (VR) is a service that uses deep learning algorithms to identify objects and other content in an image. In this hands-on lab, you will use Watson VR to upload and classify images.

## Objectives

After completing this lab, you will be able to:

1. Access IBM Cloud
2. Add resources to your IBM Cloud account
3. Add services to your IBM Cloud account
4. Create a project in Watson Studio
5. Analyze images using Watson VR

**Steps needed to be followed**

## Sign up for IBM Cloud

1. Go to: [Create a free account on IBM Cloud](https://cloud.ibm.com/registration)
2. In the **Email**box, enter your email address and then click the arrow.
3. When your email address is accepted, enter your **First Name**, **Last Name**, **Country or Region**, and create a **Password**.
4. **Note:** To get enhanced benefits, please sign up with your company email address rather than a free email ID like Gmail, Hotmail, etc.

If you would like IBM to contact you for any changes to services or new offerings, then check the box to accept the option to be notified by email.

Then click **Create Account**to create your IBM Cloud account.

## Confirm your email address

1. An email is sent to the address that you signed up with.
2. Check your email, and in the email that was sent to you, click **Confirm Account**.
3. You will receive notification that your account is confirmed.

Click **Log In**, and you will be directed to the IBM Cloud Login Page.

## Login to your IBM Cloud account

1. On the [Log in to IBM Cloud](https://cloud.ibm.com/login) page, in the ID box, enter your email address and then click **Continue**.
2. In the **Password**box, enter your password, and then click **Log in**.

## Create a project

## Scenario:

To manage all the resources and services that you are working with, you should create a Watson Studio Project. You will begin by creating an empty project, and then adding the resources and services that you need.

## Create an empty project

1. On the Watson Studio Welcome page, click **Create a project**or **New Project+** in Recent Project section.
2. On the New project page, enter a **Name**and **Description**for your project.
3. You must define storage for your project before you can create it. Under **Select storage service**, click **Add**
4. The Cloud Object Storage page will be opened in a new tab. On the Cloud Object Storage page, verify that **Lite**is selected, and then click **Create**. If it successfully created, this page will be closed automatically.
5. On the New project page, under **Define storage**, click **Refresh.**
6. The cloud object created in the previous step will be assigned and then click **Create**.

## Add a Watson VR Service instance

1. To add services to the project, click **Add to project**
2. In the Choose asset type box, click **Visual Recognition**.
3. In the Associate a service box, click **here**.
4. On Associate Service page, **make sure there is no active filter**, hence, it shows all existing services. This is the tricky part because I do not have pictures for an active filter. If there is an active filter, there will be **a black bubble with the number of the active filters** in the red box area. **Close all the active filter by clicking the cross button.** In this case, it is still no service
5. On the Visual Recognition page, select the region closest to you, verify that the **Lite**plan is selected, and then click **Create**.

## Analyze images with Watson VR

Now you can see all the built-in image classification models that IBM Watson provides! Let’s try the General model.

To analyze your images, on the Models page, under **Prebuilt Models**, in the **General**box, click **Test**.

1. On the General page, click the **Test**tab.
2. To upload images, on the Test tab, click **Browse**.
3. Select the images you want to upload and then click **Open**.
4. Once you have uploaded your images, Watson Studio Visual Recognition will tell you what it thinks it found in your images! Beside each class of object (or color, age, etc.), it gives you a confidence score (between 0 and 1) showing how confident it is that it found that particular object or feature in your image (0 for lowest confidence and 1 for highest confidence).
5. Use the checkboxes on the left to filter the images. In this example, only images in which Watson VR has detected **beige color** are displayed.
6. Use the **Threshold**slider to only display images in which Watson VR has at least 90% confidence of the beige color.

Sample code in python:

**import** json

**from** ibm\_watson **import** VisualRecognitionV3

**from** ibm\_cloud\_sdk\_core.authenticators **import** IAMAuthenticator

authenticator = IAMAuthenticator('IBM API KEY')

visual\_recognition = VisualRecognitionV3(

version='2018-03-19',

authenticator=authenticator)

visual\_recognition.set\_service\_url('IBM URL')

**with** open('./test.jpg', 'rb') **as** images\_file:

classes = visual\_recognition.classify(images\_file=images\_file,threshold='0.6',classifier\_ids='food').get\_result()

print(json.dumps(classes, indent=2))