IBM

developer.ibm.com/**code**

# Watson Visual Recognition

Lecture deck

# What is Watson Visual Recognition ?

- A cloud based service that uses deep learning algorithms to analyze images for scenes, objects, faces, and other content.
  - The response includes keywords that provide information about the content

- Major functionality
  - Image classify with keywords based on image recognized
    - Provides a default classifier that recognizes a wide range of images
    - Train your own classifiers with your images based on your own business scenarios
  - Detect human faces giving age range and gender

- Tooling
  - A web based tool for classifying images and training your own classifiers

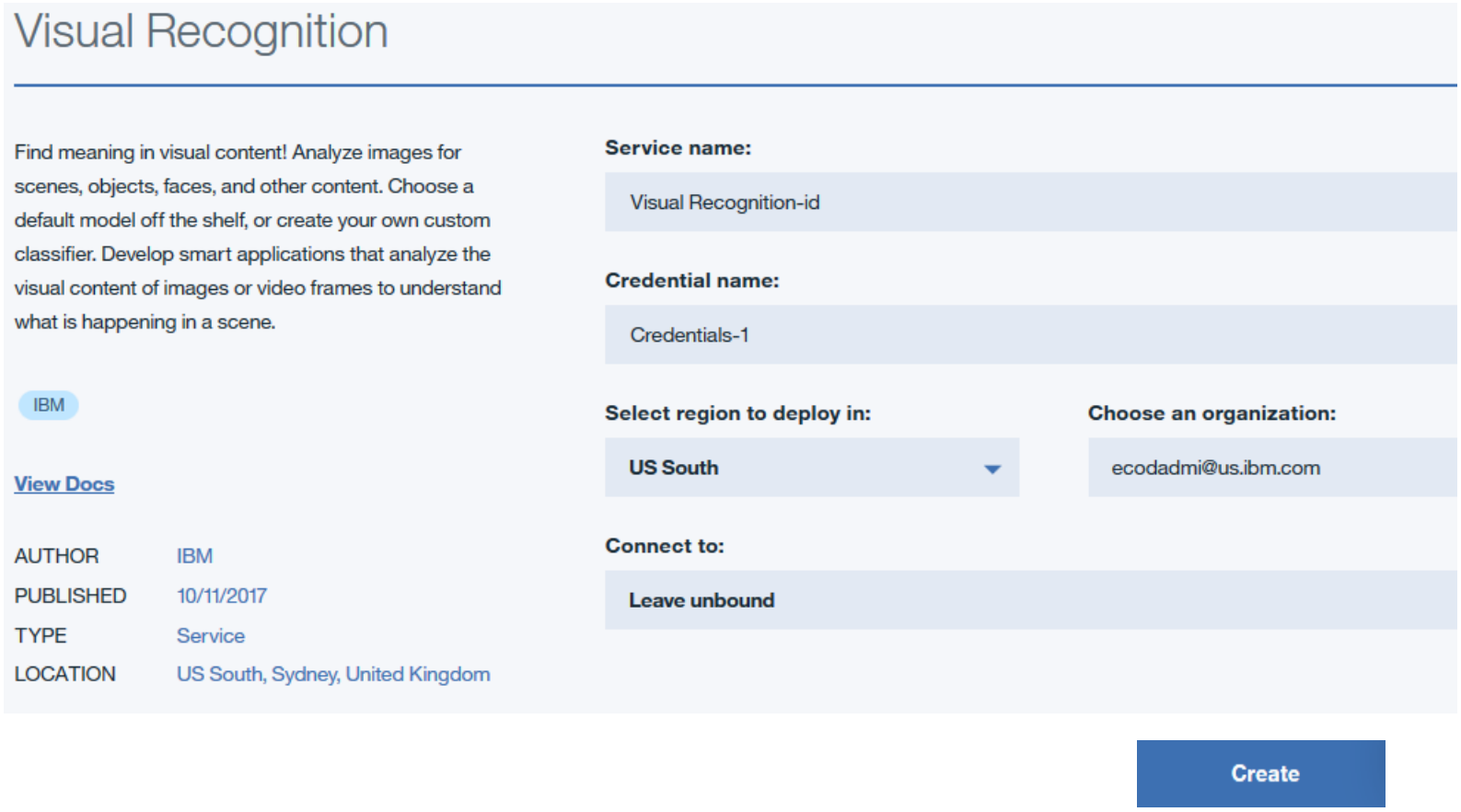
# Typical use cases

- **Manufacturing:** Use images from a manufacturing setting to make sure products are being positioned correctly on an assembly line.
- **Visual Auditing**: Look for visual compliance or deterioration in a fleet of trucks, planes, or windmills in the field,
- **Insurance:** Rapidly process claims by using images to classify claims into different categories.
- **Social listening**: Use images from your product line or your logo to track buzz about your company on social media.
- **Social commerce**: Use a house image to find similar homes that are for sale.
- **Retail:** Take a photo of a favorite outfit to find stores with those clothes in stock or on sale,.
- **Education**: Use pictures to find educational material on similar subjects. **Public safety:** Automated, real-time video stream analysis to include targeted observations such as facial recognition and automated licence-plate reading

# Getting started

- The Visual Recognition is available in Bluemix and should be provisioned from the catalog
- The API is REST based w/a public endpoint
  - For authentication the service uses an api key that must be passed as a param to each request
- Each service instance has it's own credentials
  - Credentials are available in the Bluemix console
  - For Bluemix apps bound to a service instance
    - Credentials available as JSON in the VCAP_SERVICES env var
- The service has a deployed demo app that can be run and then cloned from GitHub

# Provisioning an instance from the IBM Cloud Catalog



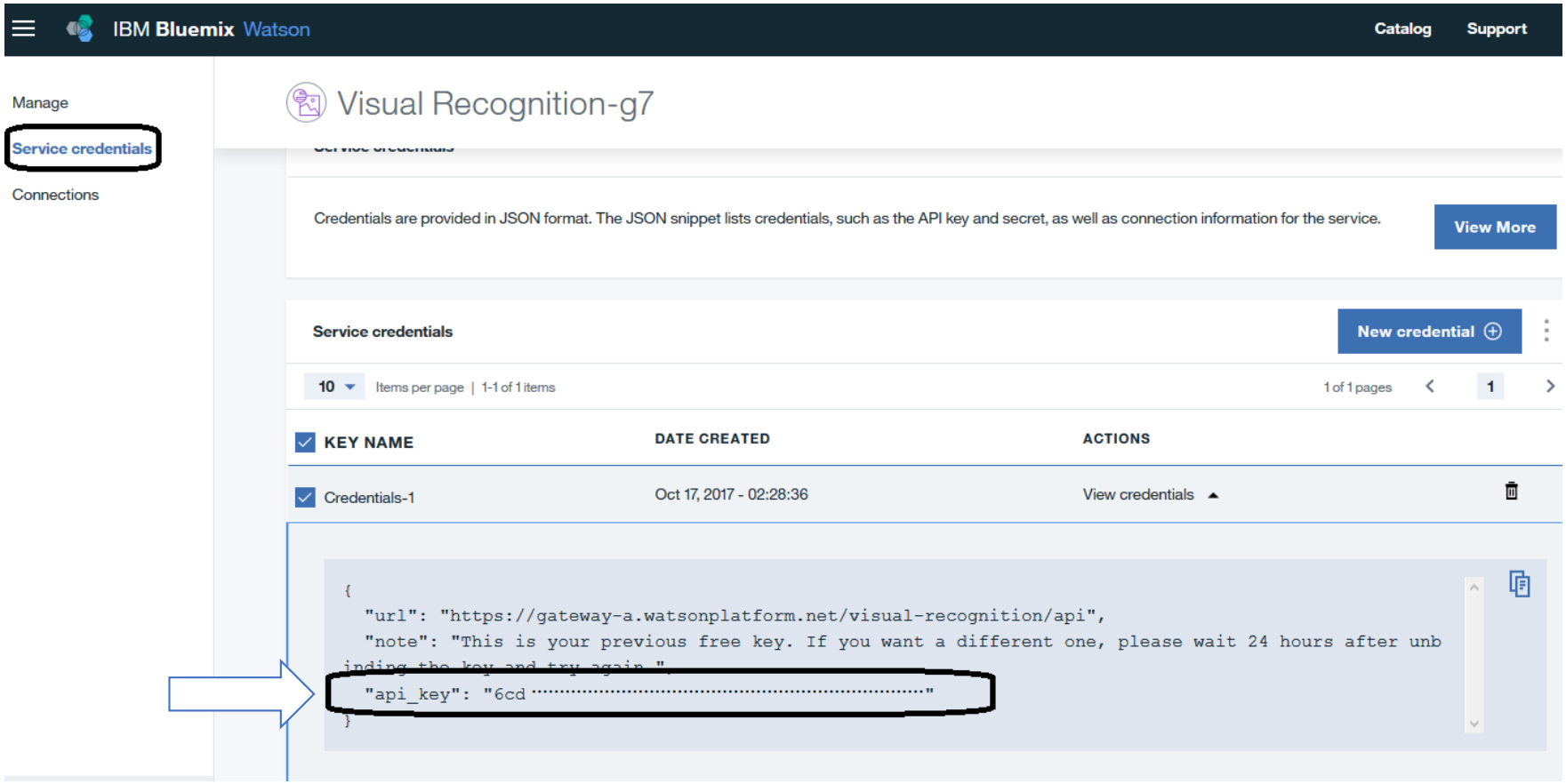
# Available plans

## Pricing Plans

Monthly prices shown are for country or region: United States

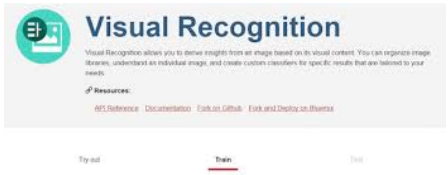
| | PLAN | FEATURES | PRICING |
|---|---|---|---|
| ✓ | **Free** | **250 Events (images) per day per Bluemix Account**<br>1 Instance per Bluemix Organization | Free |
| | | The Visual Recognition Free Plan offers 250 Events (images) per day per Bluemix Account for image tagging and face detection. | |
| | **Standard** | **Image Tagging Events Pay per Use**<br>Face Detection Events Pay per Use<br>Training Events Pay per Use<br>Custom Tagging Events Pay per Use | $0.002 USD/Event<br>$0.004 USD/Event<br>$0.10 USD/Event<br>$0.004 USD/Event |

# After provisioning the API KEY is available in the IBM Cloud console



# Using Visual Recognition w/o training

**Using the default classifier**

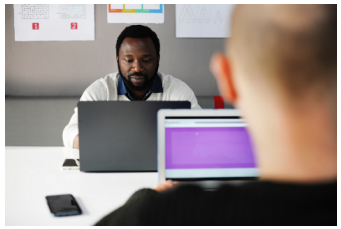


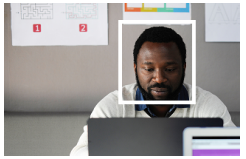

| Classes | Score |
|---|---|
| gyrfalcon | 0.97 |
| falcon | 0.97 |
| hawk | 0.97 |
| bird of prey | 0.97 |
| bird | 0.97 |
| animal | 0.97 |
| black color | 0.79 |
| ash grey color | 0.74 |

Type Hierarchy

/animal/bird/bird of prey/hawk/falcon /gyrfalcon

**Face detection**





| Faces | Score |
|---|---|
| age 35 - 44 | 0.40 |
| male | 0.99 |

# Using the default classifier

**Input image**

**REST call – GET or POST request**

https://gateway-a.watsonplatform.net/visual-recognition/api/v3/classify?api_key={api-key}
&url={url of image}&version=2016-05-20

**Result**


```
{
 "custom_classes": 0,
 "images": [
  {
   "classifiers": [
    {
     "classes": [
      {
       "class": "banana",
       "score": 0.562,
       "type_hierarchy": "/fruit/banana"
      },
      {
       "class": "fruit",
       "score": 0.788
      },
      {
       "class": "diet (food)",
       "score": 0.528,
       "type_hierarchy": "/food/diet (food)"
      },
      {
       "class": "food",
       "score": 0.528
      },
      {
       "class": "honeydew",
       "score": 0.5,
       "type_hierarchy": "/fruit/melon/honeydew"
      },
      {
       "class": "melon",
       "score": 0.501
      },
      {
       "class": "olive color",
       "score": 0.973
      },
      {
       "class": "lemon yellow color",
       "score": 0.789
      }
     ],
     "classifier_id": "default",
     "name": "default"
    }
   ],
   "image": "fruitbowl.jpg"
  }
 ],
 "images_processed": 1
}
```

# Face detection

**Input image**

**REST call – GET or POST request**

https://gateway-a.watsonplatform.net/visual-recognition/api/v3/detect_faces?api_key={api_key}
&url={url of image}&version=2016-05-20

**Result**

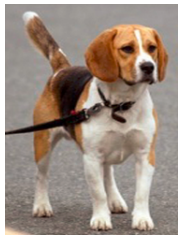

```
{
  "images": [{
    "faces": [{
      "age": {
        "max": 44,
        "min": 35,
        "score": 0.446989
      },
      "face_location": {
        "height": 159,
        "left": 256,
        "top": 64,
        "width": 92
      },
      "gender": {
        "gender": "MALE",
        "score": 0.99593
      },
      "identity": {
        "name": "Barack Obama",
        "score": 0.970688,
        "type_hierarchy": "/people/politicians/democrats/barack
obama"
      }
    }],
    "image": "prez.jpg"
  }],
  "images_processed": 1
}
```

# Using Visual Recognition – Custom classifiers

**Step 1 – Train your custom classifier**

**Your custom classifier**

**Step 2 – Use your custom classifier**

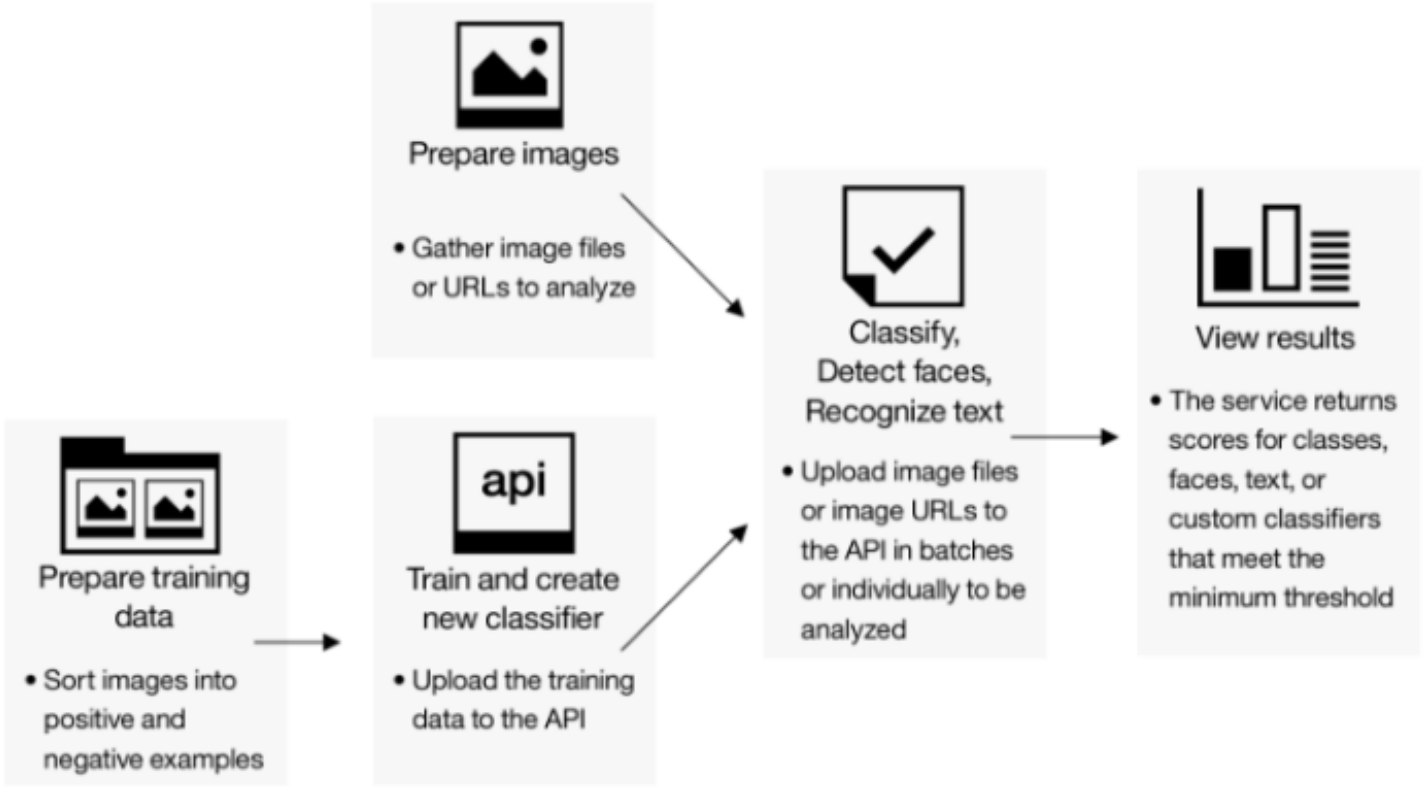


**Your custom classifier**

**+**

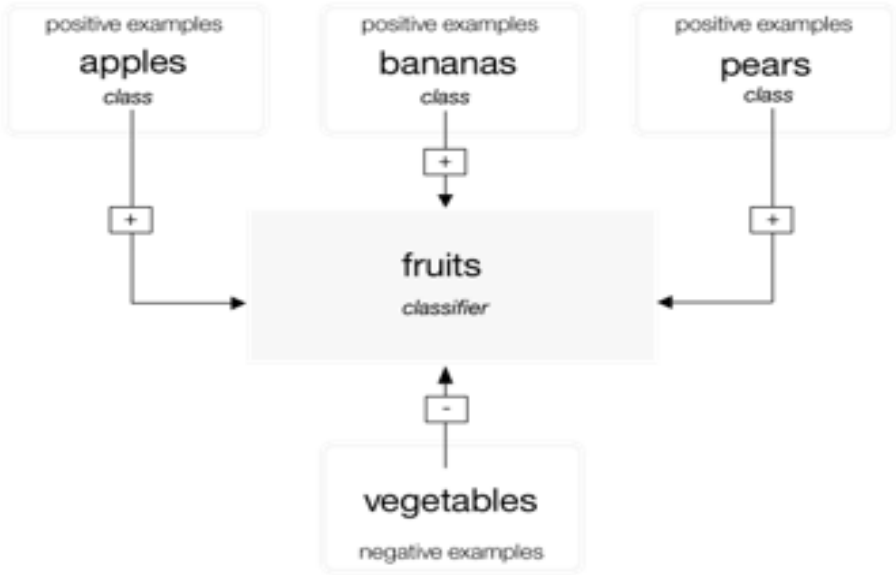
| Classes | Score |
|---|---|
| beagle | 0.98 |

# Training and using  a custom classifier



# Structure of the training data

- A custom classifier is a group of classes that are trained against each other.
  - This allows you to create a multi-faceted classifier that can identify highly specialized subjects, while also providing a score for each individual class.
- You upload separate compressed (.zip) files of positive examples for each class
  - . For example, to create a classifier called **fruit**, you might upload a .zip file of images of pears, a .zip file of images of apples, and a .zip file of images of bananas in a single training call
- You also provide a .zip file of negative examples in the same training call to further hone your classifier.
  - Negative example files are not used to create a class.
  - For the custom classifier fruit, for example, you might provide a .zip file with images of various vegetables



# Training tips

- A minimum of 50 images is recommended in each .zip file, as fewer than 50 images can decrease the quality of the trained classifier.
- If the quality and content of training data is the same, then classifiers that are trained on more images will generally be more accurate than classifiers that are trained on fewer images.
- The benefits of training a classifier on more images plateaus at around 5000 images
- Include approximately the same number of images in each examples file.
  - An unequal number of images can cause the quality of the trained classifier to decline.
- The accuracy of your custom classifier can be affected by the kinds of images you provide to train it.
  - Provide example images that are similar to the images you plan to analyze. For example, if you are training the classifier "tiger", your classifier might be less accurate if you provide only images of tigers in a zoo in an enclosure to train the classifier,

# Using the custom classifiers

**Input image**

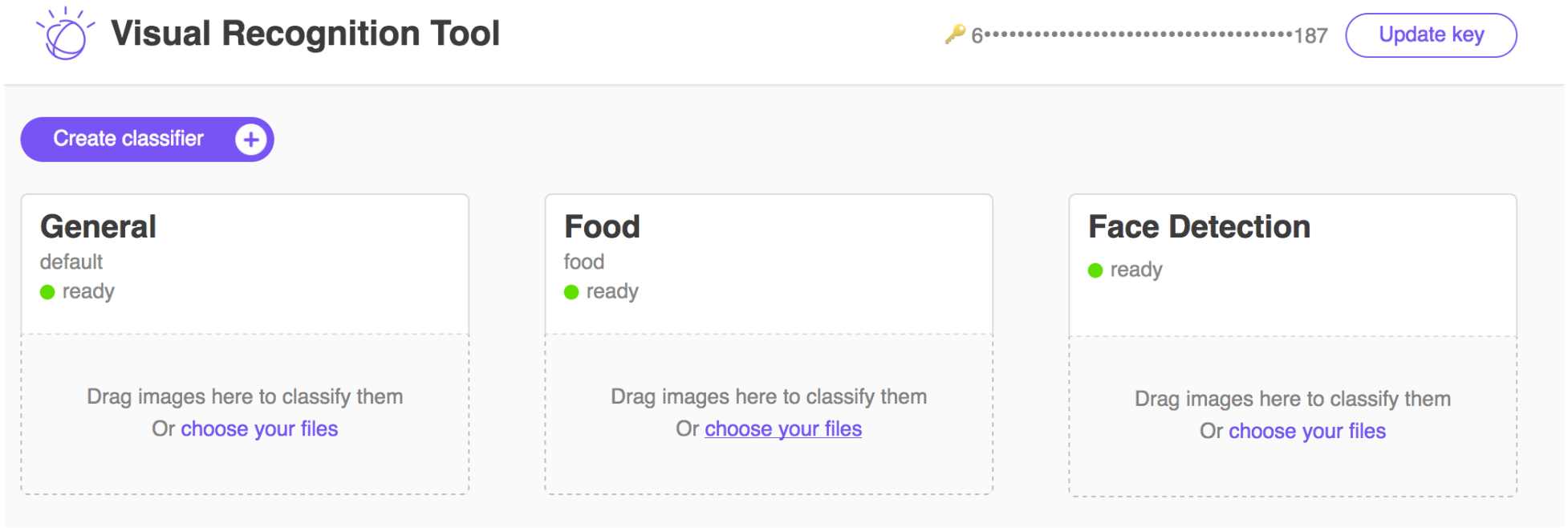
**REST call – GET or POST request indicating classifier(s) to use**

https://gateway-a.watsonplatform.net/visual-recognition/api/v3/classify?api_key={api-key}&url={url of image}&version=2016-05-20&classifier_ids=dogs_12345&threshold=0.6

**Result**


```
{
   "custom_classes": 3,
  "images": [
    {
      "classifiers": [
        {
          "classes": [
            {
              "class": "goldenretriever",
              "score": 0.611498
            }
          ],
          "classifier_id": "dogs_1009778106",
          "name": "dogs"
        }
      ],
      "resolved_url": "https://.../dogs.jpg",
      "source_url": "https://.../dogs.jpg"
    }
  ],
  "images_processed": 1
}
```

# Visual Recognition Tool




- A web based tool for classifying images, detecting faces in images and training your own classifiers
- Available at https://visual-recognition-tooling.mybluemix.net

# Useful links

- API Docs - https://www.ibm.com/watson/developercloud/visual-recognition/api/v3/
- Visual Recognition Service in Bluemix Catalog - https://console.bluemix.net/catalog/services/visual-recognition
- Demo - https://visual-recognition-demo.mybluemix.net/
- Developer Patterns
  - https://developer.ibm.com/code/journey/enrich-multi-media-files-using-ibm-watson/
- Tool: https://visual-recognition-tooling.mybluemix.net/