Microsoft

Develop computer vision solutions with Azure AI Vision

Agenda

- Analyze and manipulate images
- Create a custom vision model
- Detect and recognize faces
- Analyze video

Analyze and manipulate images

Learning Objectives

After completing this module, you will be able to:

1. Understand features and functionality of Image Analysis
2. Perform Optical Character Recognition (OCR)
3. Connect an app to Image Analysis APIs

Azure AI Vision – Image Analysis

Image analysis:

- Caption and tag generation
- Object detection
- People detection
- Optical character recognition
- Smart crop thumbnails
- Background removal
- Multi-modal embeddings
- Product recognition

Can be used as:

- Standalone **Azure AI Vision** resource
- Multi-service **Azure AI Services** resource

* Some new features are limited to specific regions

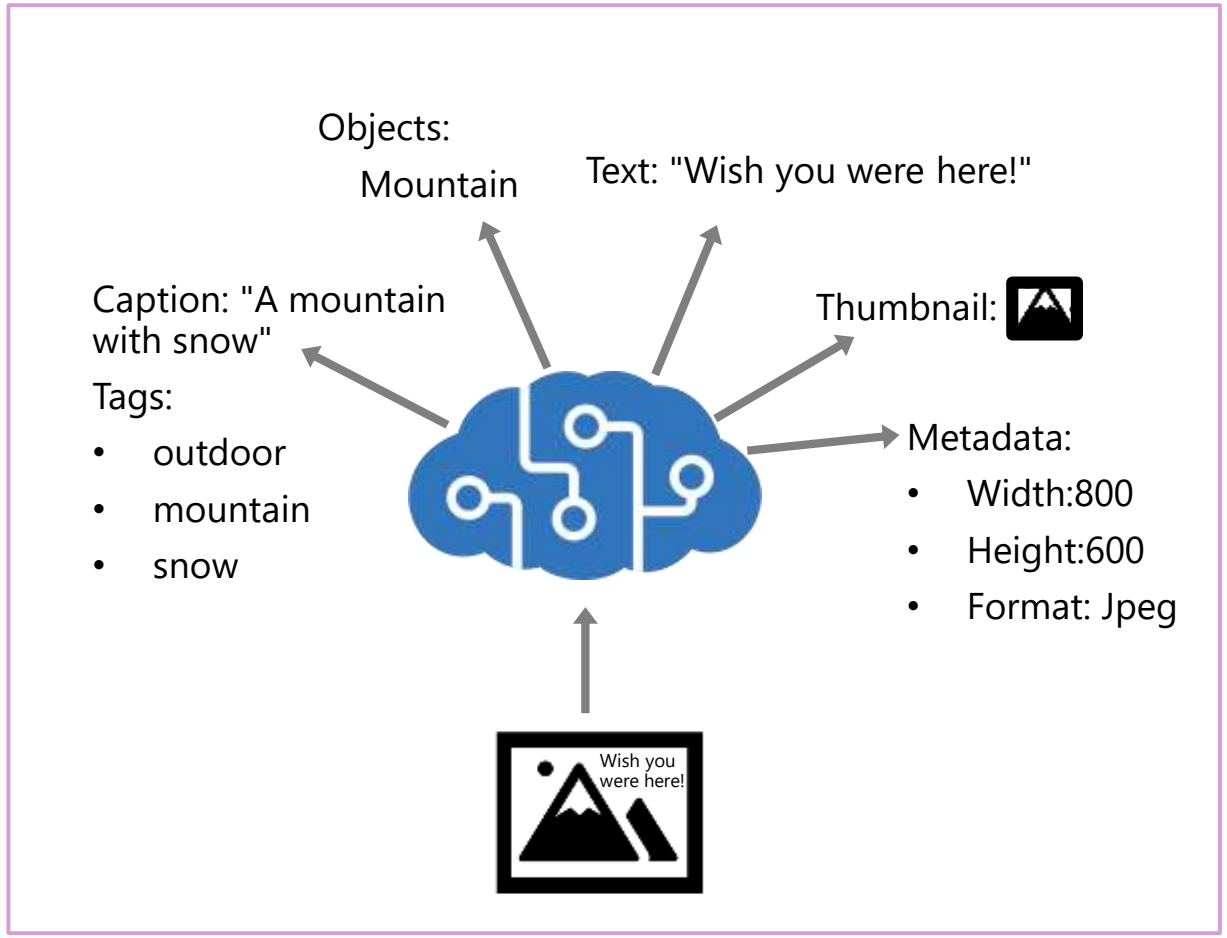


Image Analysis APIs

- Single **ImageAnalyzer** call to retrieve specified features in **ImageAnalysisFeature** enum
  - ImageAnalysisFeature.CAPTION
  - ImageAnalysisFeature.CROP_SUGGESTIONS
  - ImageAnalysisFeature.DENSE_CAPTIONS
  - ImageAnalysisFeature.OBJECTS
  - ImageAnalysisFeature.PEOPLE
  - ImageAnalysisFeature.TAGS
  - ImageAnalysisFeature.TEXT
- SDKs define the **ImageAnalyzer**, then call the **analyze()** function from it
- **Image Analyzer** provides
  - Service options: Defines endpoint and key
  - Image source: Source of the image
  - Analysis options: Which features, language, and other options for the analysis

REST

```
https://<endpoint>/computervision/imageanalysis:analyze?
features=caption,people&model-version=latest&
language=en&api-version={version}
```

C#

```
using var analyzer = new ImageAnalyzer(
          serviceOptions,
          imageSource,
          analysisOptions);

var result = analyzer.Analyze();
```

Python

```
image_analyzer = sdk.ImageAnalyzer(
          service_options, # Defines endpoint and key
          vision_source,
          analysis_options)

result = image_analyzer.analyze()
```

Image Analysis Options

- Analysis options
  - Cropping aspect rations
  - Features
  - Gender neutral caption
  - Language
  - Model name
  - Model version
  - Segmentation mode

C#

```
var serviceOptions = new VisionServiceOptions(
        “{vision_endpoint}”, “{vision_key}”);
var imageSource = VisionSource.FromUrl(“{url}”);
var analysisOptions = new ImageAnalysisOptions() {
        Features = ImageAnalysisFeature.Caption |
                   ImageAnalysisFeature.Text,
        Language = "en",
        GenderNeutralCaption = true
};
```

Python

```
service_options = sdk.VisionServiceOptions(
          “{vision_endpoint}”, “{vision_key}”)

vision_source = sdk.VisionSource(url={url})
analysis_options = sdk.ImageAnalysisOptions()
analysis_options.features = (
    sdk.ImageAnalysisFeature.CAPTION |
    sdk.ImageAnalysisFeature.TEXT
)
analysis_options.language = "en"
analysis_options.gender_neutral_caption = True
```

Image Analysis Result

- Successful image analysis returns JSON (REST) or an object (SDKs)
- Results may have one or several layers of depth
  - Tags > values[ ] > name
  - Text > lines > words


```
{
    "captionResult":
    {
        "text": "a man pointing at a screen",
        "confidence": 0.4891590476036072
    },
    "tagsResult": {
         "values": [
             {
                  "name": "string",
                  "confidence": 0.0
             }
         ]
    },
    "modelVersion": "2023-02-01-preview",
    "metadata":
    {
        "width": 1038,
        "height": 692
    },
    ...
}
```

Azure AI Vision - OCR

Use **Image analysis** with TEXT feature

Vision OCR vs Document Intelligence:

- OCR: General, non-document images with smaller amounts of text. Synchronous API.
- Document Intelligence: Ideal for larger text heavy documents. Asynchronous API.

Results in JSON (REST) or object (SDK) of similar structure

```
"readResult":
  {
    "stringIndexType": "TextElements",
    "content": “Microsoft is …",
    "pages": [
        {
           "height": 945,
           "width": 1000,
           "angle": -1.099,
           "pageNumber": 1,
           "words": [
             {
               "content": “Microsoft",
               "boundingBox": [253,268,301,267,304,318,256,318],
               "confidence": 0.998,
               "span": {"offset":0,"length":10}
             },
             ...
    ],
    "lines": [
      {
        "content": "You must be the change you",
        "boundingBox": [253,267,670,262,671,307,254,318],
        "spans": [{"offset":0,"length":26}]
      },
    ...
    "styles": [
      {
        "isHandwritten": true,
...
```

Detecting Faces with the Azure AI Vision

Learning Objectives

After completing this module, you will be able to:

1. Understand features, use cases, and responsibility of the Azure AI Vision Face API

2. Use the Face API in an app

Options for Face Detection, Analysis, and Recognition

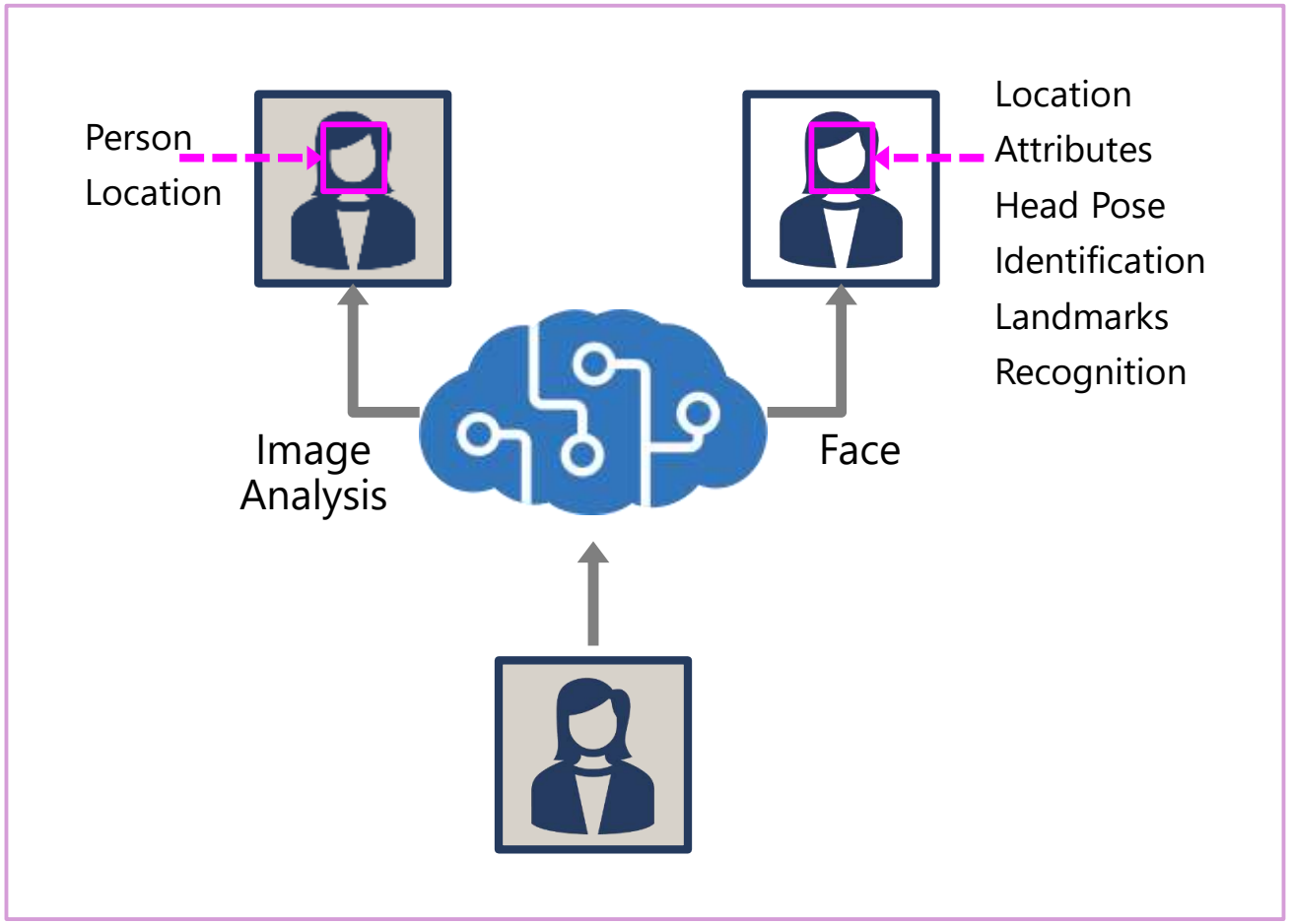
Image Analysis

- People detection
- Only location provided

Face Service

- Face detection
- Comprehensive facial feature analysis
- Face comparison and identification*
- Facial recognition*

** Require Limited Access approval*



Considerations for Face Detection and Facial Recognition

Principles of responsible AI apply to all kinds of application, but systems that rely on facial data can be particularly problematic. As a safeguard for responsible AI usage, facial recognition, identification, verification, and comparison is behind a Limited Access policy, requiring users to be approved my Microsoft before enabling these features.

Data privacy and security

Systems based on facial data should protect individual privacy, ensuring that personally identifiable data is not accessed inappropriately

Transparency

Users should be informed about how their image will be used, and who will have access to it.

Fairness and Inclusiveness

Facial recognition should not be used in a manner that is prejudicial to individuals based on their appearance, or to unfairly target individuals

The Face Service

Face detection

Face attribute analysis

Facial landmark location

- Nose, eyes, mouth, …

Face comparison*

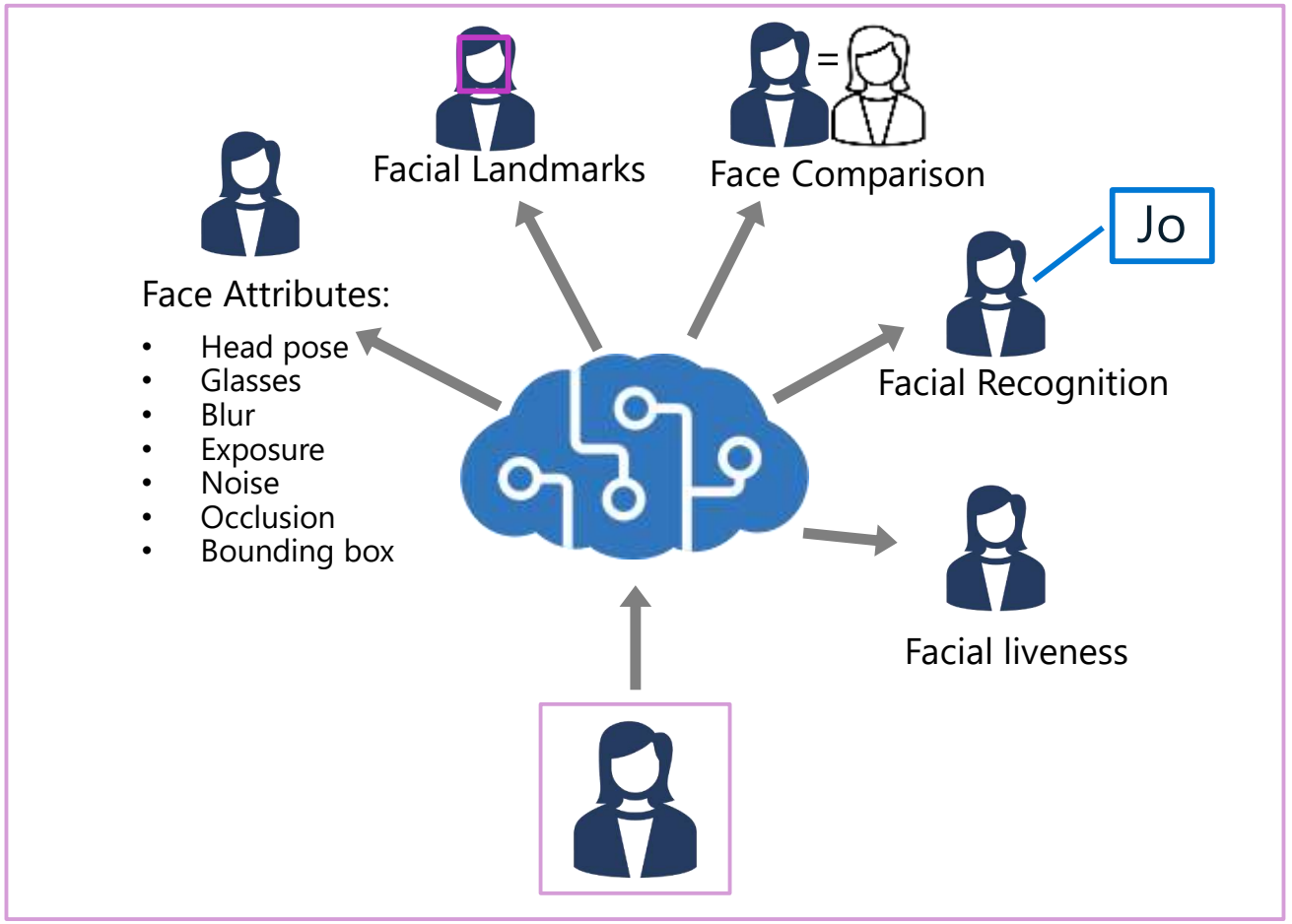
Facial recognition and identification*

Facial liveness*

Can be used as:

- Standalone **Face** resource
- Multi-service **Azure AI Services** resource

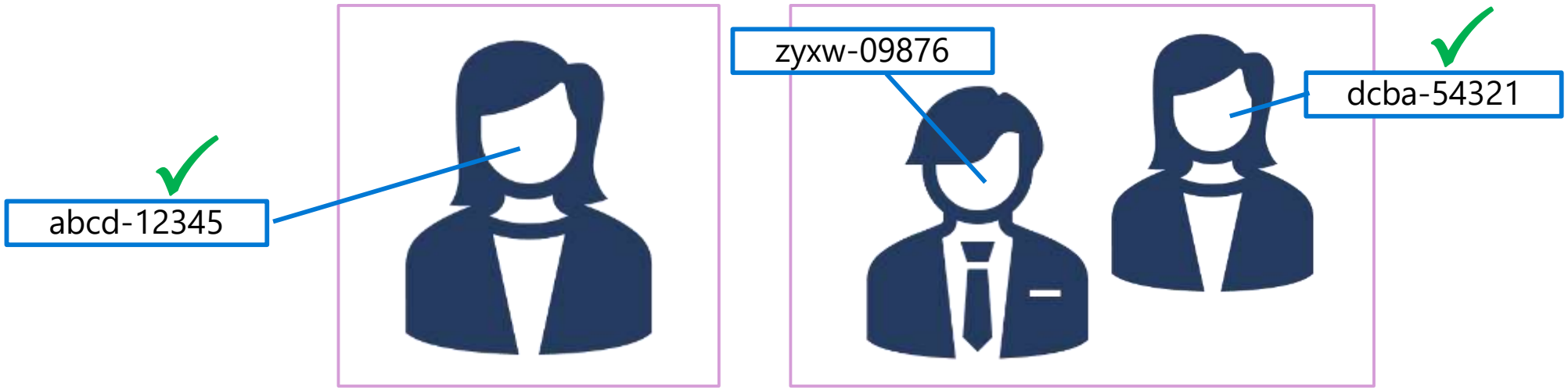
** Require Limited Access approval*



Detected Face Identification

Every *detected* face is assigned an anonymous ID

- Retained in your service resource for 24 hours
- Can be used to compare faces in multiple images
  - *Verify* faces to determine if they're the same individual
  - *Find similar* faces to identify faces with similar features



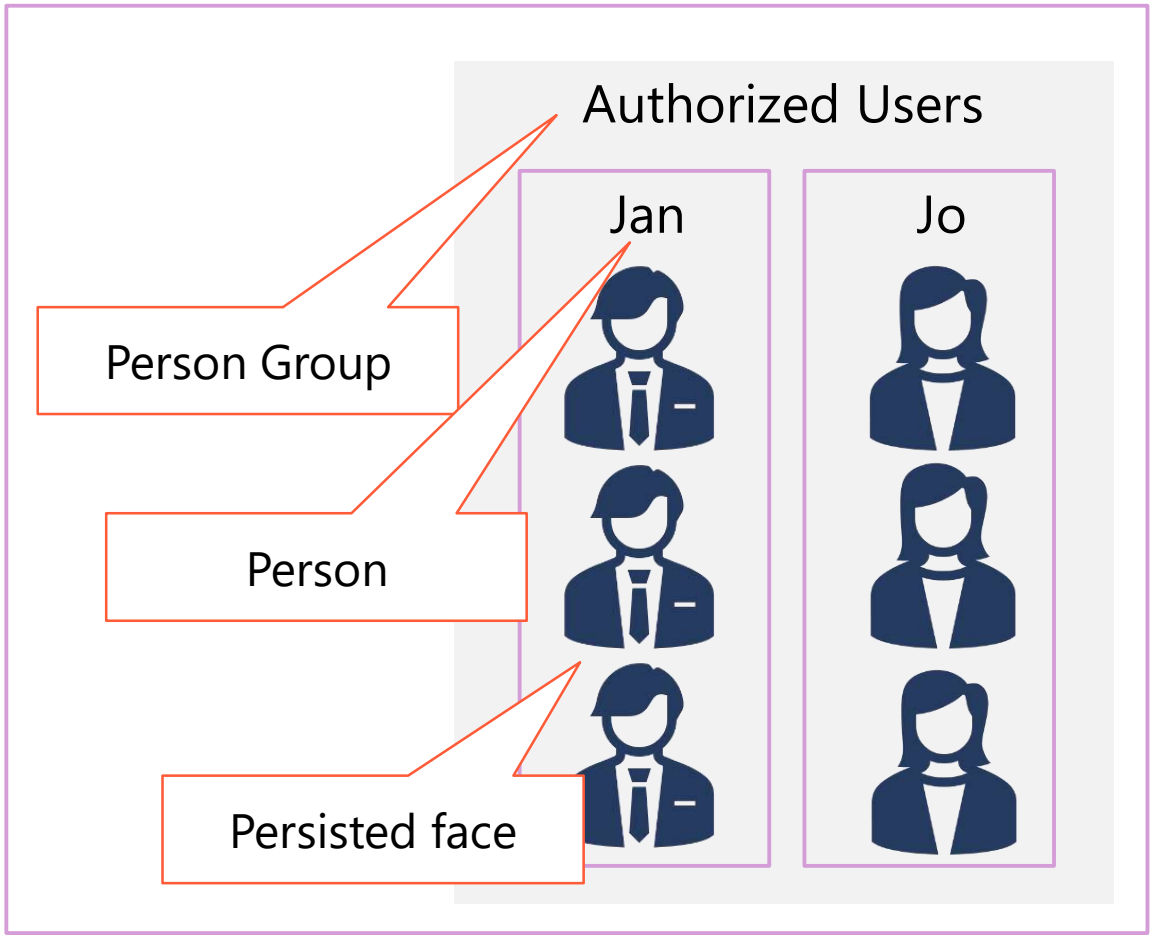
Persisted Face Recognition

Train a facial recognition model using face images

1. Create a **Person Group** for the people you want to identify
2. Add a **Person** for each individual
3. Add multiple detected **Faces** to each person
   - These become *persisted* faces
4. Train the model

Use the model for facial recognition

- *Identify* an individual person
- *Verify* the face of an individual person
- *Find* similar faces to a persisted face



Face Detection with Azure AI Vision

Use the **Face** endpoint, specifying **Faces** as a visual feature

Note: Most features like recognition and identification are not enabled for new users

Optional request parameters	
returnFaceId	recognitionModel
returnFaceLandmarks	returnRecognitionModel
returnFaceAttributes	detectionModel

```
Request: https://{endpoint}/face/v1.0/detect[?options]
Body: {"url": "http://path-to-image}

Response:
[
    {
        "faceId": "c5c24a82-6845-4031-9d5d-978df9175426",
        "recognitionModel": "recognition_03",
        "faceRectangle": {
            "width": 78,
            "height": 78,
            "left": 394,
            "top": 54
        },
        "faceLandmarks": {
            "pupilLeft": {
                "x": 412.7,
                "y": 78.4
            },
            "pupilRight": {
                "x": 446.8,
                "y": 74.2
            },
            ...
```

Custom vision models with Azure AI Vision

Learning Objectives

After completing this module, you will be able to:

1. Understand use cases of custom Vision models
2. Label data in Azure ML for both image classification and object detection
3. Use a custom vision model in an app

Two types of custom vision models

Azure AI Custom Vision (previous service)

- Portal: **customvision.ai**
- Base model:
  - Convolutional neural network (CNN)
- Tasks:
  - Image classification
  - Object detection
- Labeling:
  - Customvision.ai
- Minimum training data needed:
  - 15 images per category
- Training data storage
  - Uploaded to Custom Vision service

Custom AI Vision models (new Florence model)

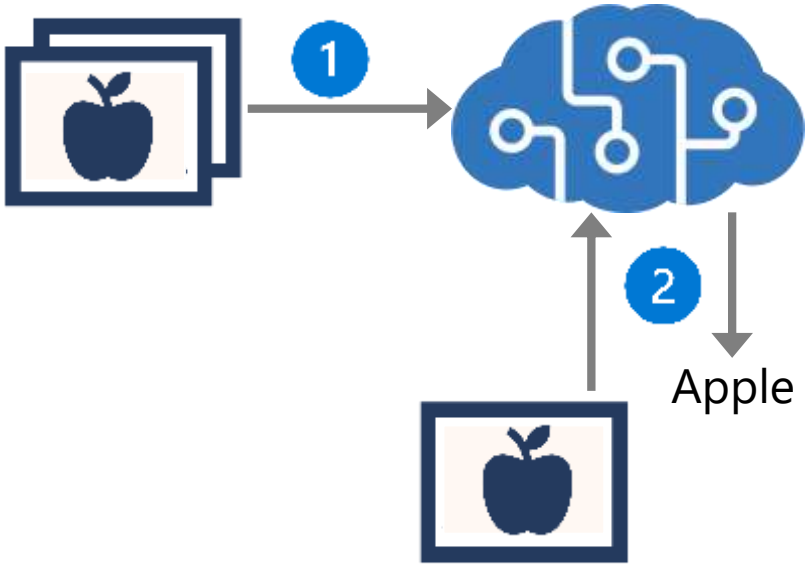
- Portal: **Vision Studio**
- Base model:
  - Transformer (multi modal)
- Tasks:
  - Image classification
  - Object detection
  - Product recognition
- Labeling:
  - AML Studio or COCO file
- Minimum training data needed:
  - 2-5 images per category
- Training data storage
  - In user's blob storage account

Custom Azure AI Vision model

Train custom models with your own images

- Upload your images
- Label your images
- Train your model
- Query your model with new images to predict labels

Labeling data lives in COCO file



1. Use your own images to train a model
2. Use the model to predict labels for new images

About COCO files

JSON file with specific fields

- images
- annotations
- categories

Defines labeling data from Azure ML project

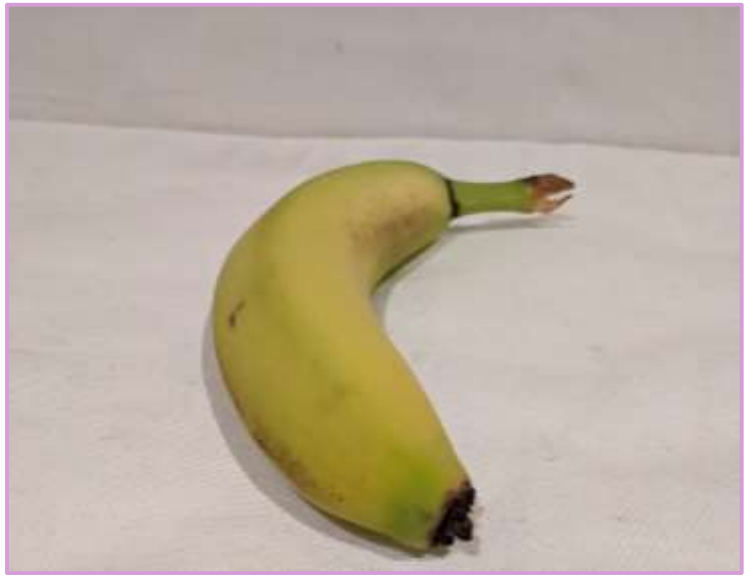
```
{
  "images": [
    {
      "id": 1,
      "width": 500,
      "height": 828,
      "file_name": “file.jpg",
      "absolute_url": “{url}"
    },
    ...
  ],
  "annotations": [
    {
      "id": 1,
      "category_id": 7,
      "image_id": 1,
      "area": 0.407,
      "bbox": [
        0.02663142641129032,
        0.40691584277841153,
        0.9524163571731749,
        0.42766634515266866
      ]
    },
    ...
  ],
  "categories": [
    {
      "id": 1,
      "name": "apple"
    },
...
```

What is Image Classification?

Train a model to predict the class label for the image

In other words, what is this a picture of?

Apple



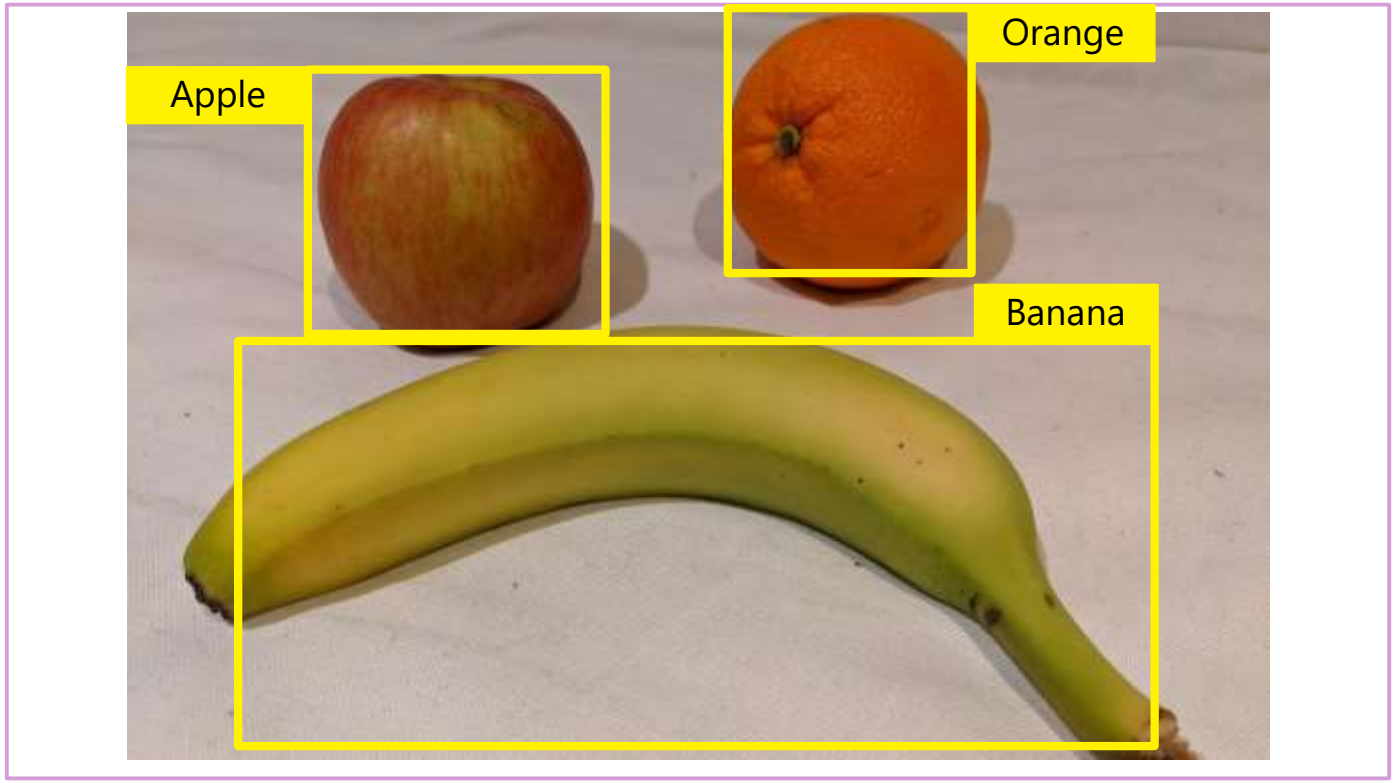
Banana

Orange

What is Object Detection?

Train a model to detect and locate specific classes of object in images

In other words, what objects are in this image, and where?

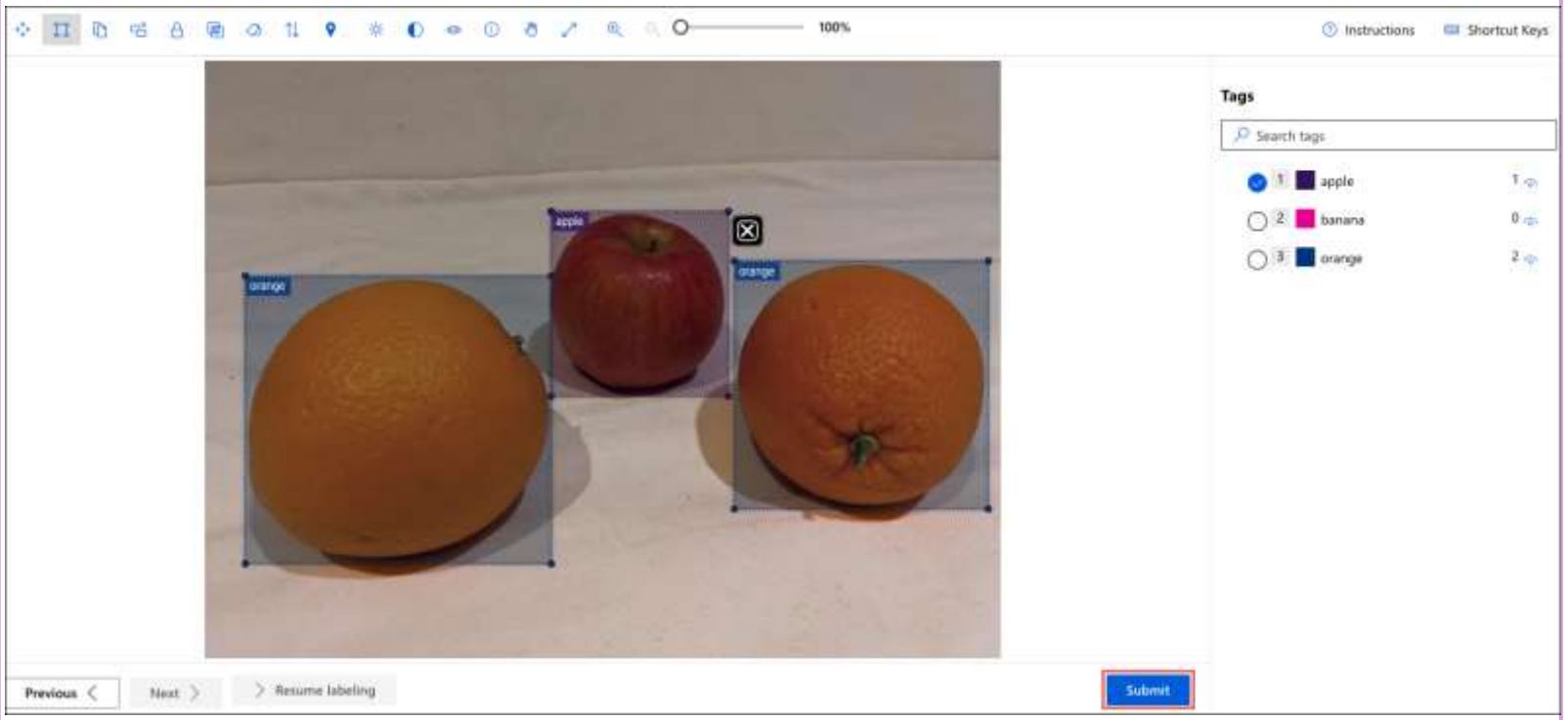


Train a custom model

Use the Azure Vision Studio

1. Create a custom model project, or retrieve an existing one
2. Select your resource, if necessary
3. Add your dataset(s) and specify model type
    1. Image classification: Assign label to each image
    2. Object detection: Define bounding boxes for objects in each image
4. Create your Azure ML project and label your images (if no COCO file exists)
5. Add COCO file (from Azure ML project or previously built)
6. Train a new model

Use the Azure ML Data Labeling project

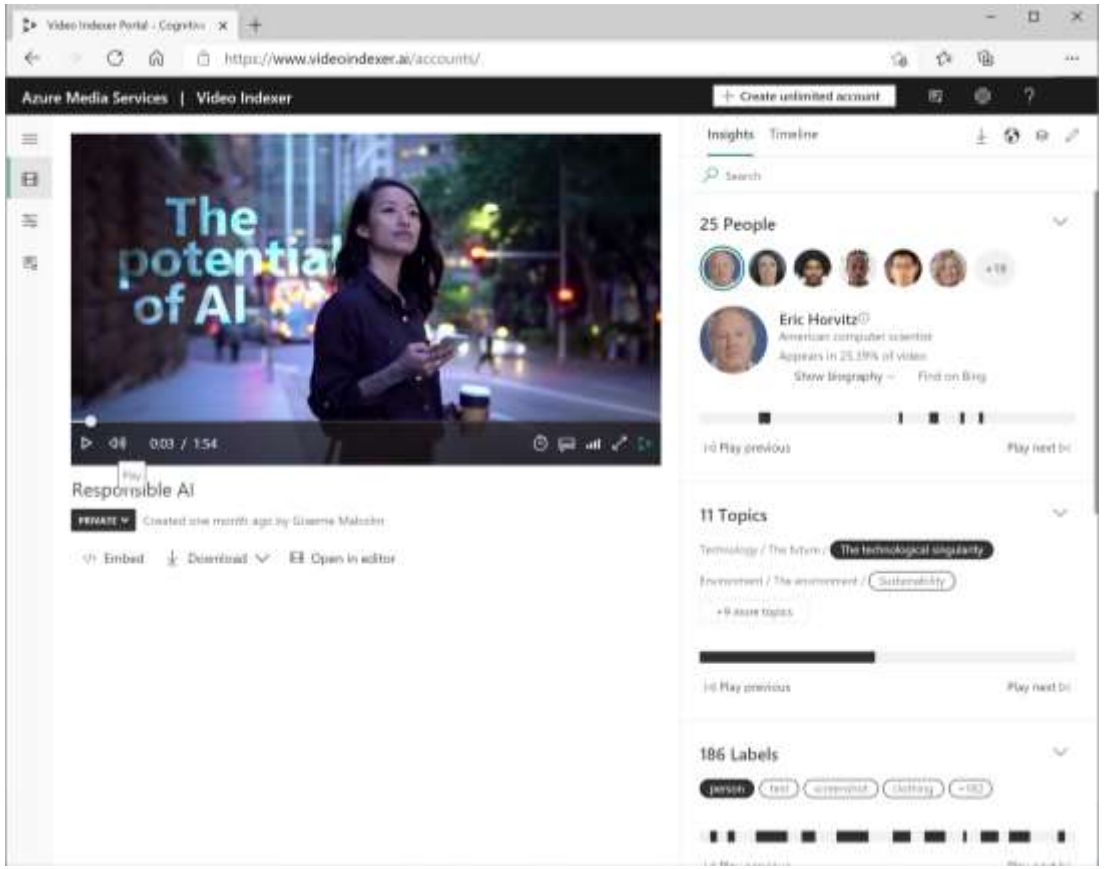


Analyzing Videos

Video Indexer

Video analysis:

- Facial recognition (limited access)
- Optical character recognition
- Speech transcription
- Topics
- Sentiment
- Labels
- Content moderation
- Scene segmentation



Custom Insights

Pre-defined models for recognizing language, well-known celebrities, brands, …

Create your own models for:

People

Train facial recognition* from sample images

Language

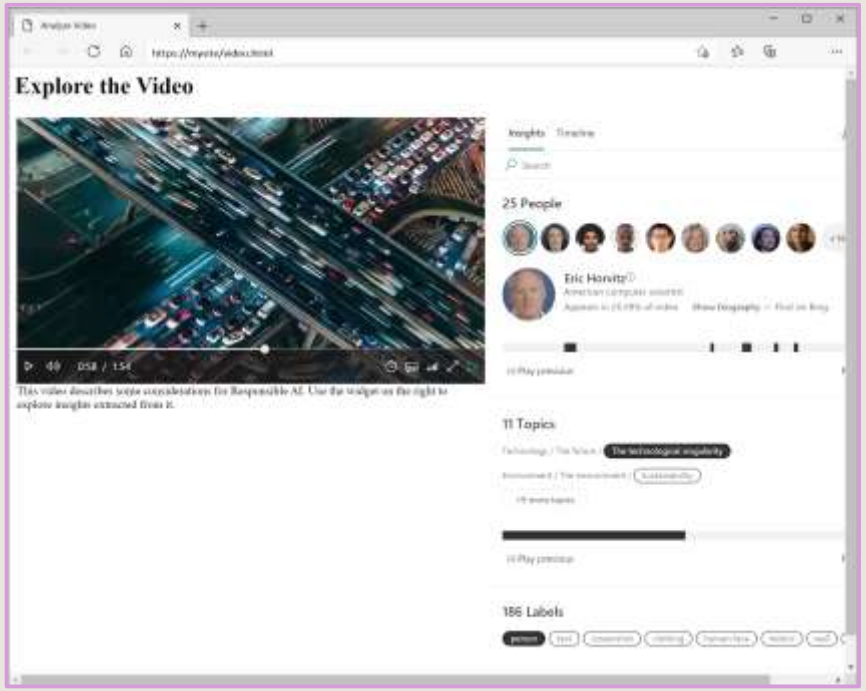
Recognize industry/ organization-specific terms

Brands

Recognize brand names for products, companies, projects,..

* Facial recognition and celebrity identification requires Limited Access approval, in accordance with our Responsible AI Standard

Video Indexer Widgets and API



Share insights and analysis in web pages

https://api.videoindexer.ai.../Videos?accessToken=$token


```
{
  "results": [
    {
      "accountId":  "1234abcd-9876fghi-0156kihb-00123",
      "id":  "a12345bc6",
      "name":  "Responsible AI",
      "description":  "Microsoft Responsible AI video",
      "created":  "2021-01-05T15:33:58.918+00:00",
      "lastModified":  "2021-01-05T15:50:03.123+00:00",
      "lastIndexed":  "2021-01-05T15:34:08.007+00:00",
      "processingProgress":  "100%",
      "durationInSeconds":  114,
      "sourceLanguage":  "en-US",
    }
  ],
}
```


Automate video analysis with the REST API

Learning Path Recap

In this learning path, we:

Used Image Analysis to analyze images, extract insights, remove background, and perform OCR.

Detected faces and facial recognition.

Created custom vision models trained on your own images.

Extracted insights from videos with Azure Video Indexer.