

Analyze images

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/analyze-images/1-introduction>

Introduction

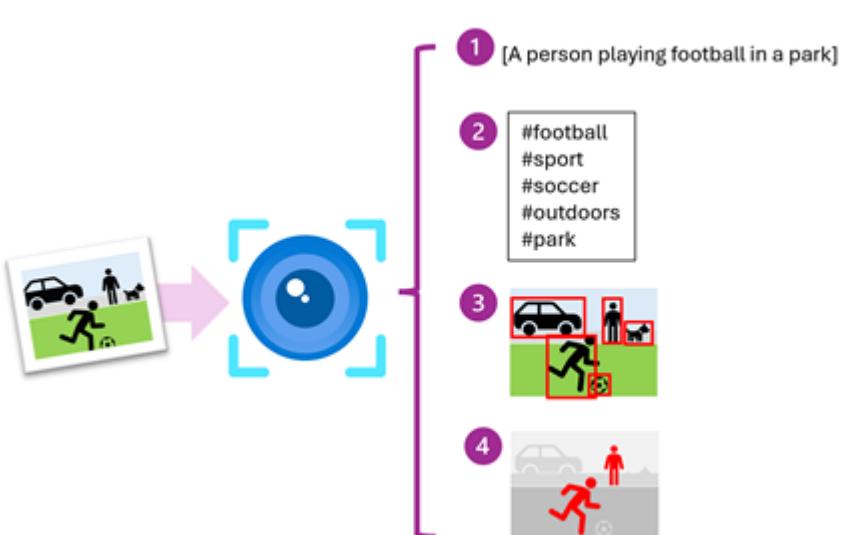
Completed

- 3 minutes

Computer vision is a branch of artificial intelligence (AI) in which software interprets visual input, often from images or video feeds. In Microsoft Azure, you can use the **Azure Vision** service to implement multiple computer vision scenarios, including:

- Image analysis
- Optical character recognition (OCR)
- Face detection and analysis
- Video analysis

In this module, we'll focus on *image analysis*, and explore how to build applications that use the Azure Vision service to analyze and extract and infer insights from images.



As shown in this conceptual diagram, the Azure Vision service provides services that you can use to analyze images and:

1. Generate a *caption* for an image based on its contents.
2. Suggest appropriate *tags* to associate with an image.
3. Detect and locate common *objects* in an image.
4. Detect and locate *people* in an image.

2. Provision an Azure Vision resource

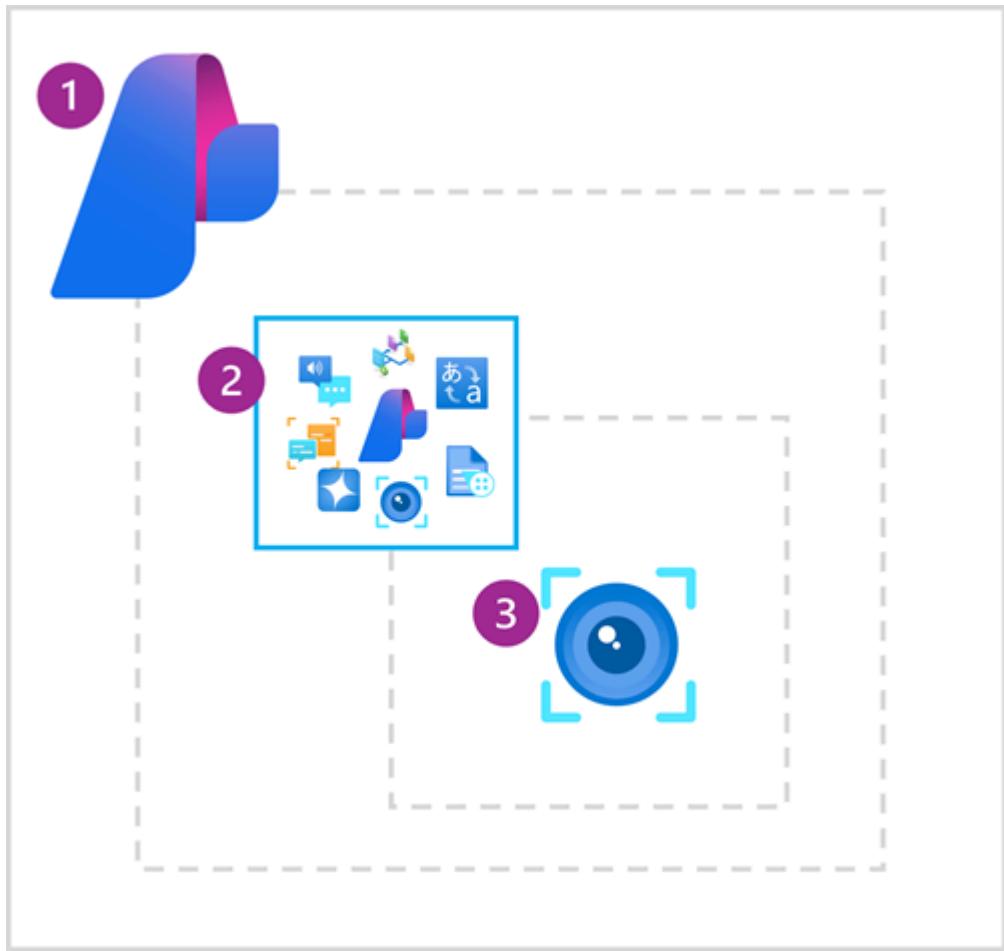
<https://learn.microsoft.com/en-us/training/modules/analyze-images/2-provision-computer-vision-resource>

Provision an Azure Vision resource

Completed

- 3 minutes

To use Azure Vision image analysis services, you need to provision an Azure Vision resource in your Azure subscription. You can choose from multiple provisioning options:



1. Create a **Microsoft Foundry** project and an associated *hub*. By default, A Microsoft Foundry hub includes a **Foundry Tools** resource, which includes Azure Vision. Microsoft Foundry projects are recommended for development of AI solutions on Azure that combine generative AI, agents, and pre-built Foundry Tools, or which involve collaborative development by a team of software engineers and service operators.
2. If you don't need all of the functionality in a Microsoft Foundry hub, you can create a **Foundry Tools** resource in your Azure subscription. You can then use this resource to access Azure Vision services and other AI services through a single endpoint and key.
3. If you only need to use Azure Vision functionality, or you're just experimenting with the service, you can create a standalone **Azure Vision** resource in your Azure subscription. One benefit of this approach is that the standalone service provides a free tier that you can use to explore the service at no cost.

Tip

If you're unfamiliar with Microsoft Foundry and Foundry Tools, consider completing the [Plan and prepare to develop AI solutions on Azure](#) module.

Connecting to your resource

After you've deployed your resource, you can use the [Azure Vision REST API](#) or a language-specific SDK (such as the [Python SDK](#) or [Microsoft .NET SDK](#)) to connect to it from a client application.

Every Azure Vision resource provides an *endpoint* to which client applications must connect. You can find the endpoint for your resource in the Azure portal, or if you're working in a Microsoft Foundry project, in the Microsoft Foundry portal. The endpoint is in the form of a URL, and typically looks something like this:

```
https://<resource_name>.cognitiveservices.azure.com/
```

To connect to the endpoint, client applications must be authenticated. Options for authentication include:

- **Key-based authentication:** Client applications are authenticated by passing an authorization key (which you can find and manage in the portal).
- **Microsoft Entra ID authentication:** Client applications are authenticated by using a Microsoft Entra ID token for credentials that have permission to access the Azure Vision resource in Azure.

When developing and testing an application, it's common to use key-based authentication or Microsoft Entra ID authentication based on your own Azure credentials. In production, consider using Microsoft Entra ID authentication based on a managed identity for your Azure application or use Azure Key Vault to store authorization keys securely.

Note

When using a Foundry Tools resource in a Microsoft Foundry project, you can use the Microsoft Foundry SDK to connect to the project using Microsoft Entra ID authentication, and then retrieve the connection information for your Foundry Tools resource, including the authorization key, from the project.

3. Analyze an image

<https://learn.microsoft.com/en-us/training/modules/analyze-images/3-analyze-image>

Analyze an image

Completed

- 3 minutes

After connecting to your Azure Vision resource endpoint, your client application can use the service to perform image analysis tasks.

Note the following requirements for image analysis:

- The image must be presented in JPEG, PNG, GIF, or BMP format.
- The file size of the image must be less than 4 megabytes (MB).
- The dimensions of the image must be greater than 50 x 50 pixels.

Submitting an image for analysis

To analyze an image, you can use the [Analyze Image](#) REST method or the equivalent method in the SDK for your preferred programming language, specifying the visual features you want to include in the analysis.

```
from azure.ai.vision.imageanalysis import ImageAnalysisClient
from azure.ai.vision.imageanalysis.models import VisualFeatures
from azure.core.credentials import AzureKeyCredential

client = ImageAnalysisClient(
    endpoint=<YOUR_RESOURCE_ENDPOINT>,
    credential=AzureKeyCredential("<YOUR_AUTHORIZATION_KEY>")
)

result = client.analyze(
    image_data=<IMAGE_DATA_BYTES>, # Binary data from your image file
    visual_features=[VisualFeatures.CAPTION, VisualFeatures.TAGS],
    gender_neutral_caption=True,
)
```

Note

In this code example, the client app uses key-based authentication. To use Microsoft Entra ID authentication, you can use a **TokenCredential** instead of an **AzureKeyCredential**.

The code example submits the image data as a binary object (which would typically be read from an image file). You can also analyze an image based on a URL by using the **analyze_from_url** method.

Available visual features are contained in the `VisualFeatures` enumeration:

- VisualFeatures.TAGS: Identifies tags about the image, including objects, scenery, setting, and actions
- VisualFeatures.OBJECTS: Returns the bounding box for each detected object
- VisualFeatures.CAPTION: Generates a caption of the image in natural language
- VisualFeatures.DENSE_CAPTIONS: Generates more detailed captions for the objects detected
- VisualFeatures.PEOPLE: Returns the bounding box for detected people
- VisualFeatures.SMART_CROPS: Returns the bounding box of the specified aspect ratio for the area of interest
- VisualFeatures.READ: Extracts readable text

```
using Azure.AI.Vision.ImageAnalysis;

ImageAnalysisClient client = new ImageAnalysisClient(
    "<YOUR_RESOURCE_ENDPOINT>",
    new AzureKeyCredential("<YOUR_AUTHORIZATION_KEY>"));

ImageAnalysisResult result = client.Analyze(
    <IMAGE_DATA_BYTES>, // Binary data from your image file
    VisualFeatures.Caption | VisualFeatures.Tags,
    new ImageAnalysisOptions { GenderNeutralCaption = true });
```

Note

In this code example, the client app uses key-based authentication. To use Microsoft Entra ID authentication, you can use a **TokenCredential** instead of an **AzureKeyCredential**.

The code example submits the image data as a binary object (which would typically be read from an image file). You can also analyze an image based on a URL.

Available visual features are contained in the `VisualFeatures` enumeration:

- VisualFeatures.Tags: Identifies tags about the image, including objects, scenery, setting, and actions
- VisualFeatures.Objects: Returns the bounding box for each detected object
- VisualFeatures.Caption: Generates a caption of the image in natural language
- VisualFeatures.DenseCaptions: Generates more detailed captions for the objects detected
- VisualFeatures.People: Returns the bounding box for detected people
- VisualFeatures.SmartCrops: Returns the bounding box of the specified aspect ratio for the area of interest
- VisualFeatures.Read: Extracts readable text

Specifying the visual features you want analyzed in the image determines what information the response will contain. Most responses will contain a bounding box (if a location in the image is reasonable) or a confidence score (for features such as tags or captions).

Processing the response

This method returns a JSON document containing the requested information. The JSON response for image analysis looks similar to this example, depending on your requested features:

```
{  
    "apim-request-id": "abcde-1234-5678-9012-f1g2h3i4j5k6",  
    "modelVersion": "<version>",  
    "denseCaptionsResult": {  
        "values": [  
            {  
                "text": "a house in the woods",  
                "confidence": 0.7055229544639587,  
                "boundingBox": {  
                    "x": 0,  
                    "y": 0,  
                    "w": 640,  
                    "h": 640  
                }  
            },  
            {  
                "text": "a trailer with a door and windows",  
                "confidence": 0.6675070524215698,  
                "boundingBox": {  
                    "x": 214,  
                    "y": 434,  
                    "w": 154,  
                    "h": 108  
                }  
            }  
        ]  
    },  
    "metadata": {  
        "width": 640,  
        "height": 640  
    }  
}
```

4. Exercise - Analyze images

<https://learn.microsoft.com/en-us/training/modules/analyze-images/5-exercise-computer-vision>

Exercise - Analyze images

Completed

- 30 minutes

Now it's your turn to try using the Azure Vision service.

In this exercise, you use the Azure Vision SDK to develop a client application that analyzes images.

Note

To complete this lab, you need an [Azure subscription](#) in which you have administrative access.

Launch the exercise and follow the instructions.

[Launch Exercise](#)

Tip

After completing the exercise, if you've finished exploring Foundry Tools, delete the Azure resources that you created during the exercise.

5. Module assessment

<https://learn.microsoft.com/en-us/training/modules/analyze-images/6-knowledge-check>

Module assessment

Completed

- 2 minutes

6. Summary

<https://learn.microsoft.com/en-us/training/modules/analyze-images/7-summary>

Summary

Completed

- 3 minutes

In this module, you learned how to provision an Azure Vision resource and use it from a client application to analyze images.

You can use Azure Vision's image analysis capabilities in scenarios that require information extraction or inference from images. A common use case is digital asset management (DAM), in which you need to tag, catalog, and index image-based data.

To learn more about image analysis with the Azure Vision service, see the [Azure Vision documentation](#).