

Clone the repository for this course

If you have not already cloned **mslearn-ai-vision** code repository to the environment where you're working on this lab, follow these steps to do so. Otherwise, open the cloned folder in Visual Studio Code.

- ☒ 1. Start Visual Studio Code.
- ☒ 2. Open the palette (SHIFT+CTRL+P) and run a **Git: Clone** command to clone the `https://github.com/MicrosoftLearning/ai-vision` repository to a local folder (it doesn't matter which folder).
- ☐ 3. When the repository has been cloned, open the folder in Visual Studio Code.
- ☐ 4. Wait while additional files are installed to support the C# code projects in the repo.

Note: If you are prompted to add required assets to build and debug, select **Not Now**. If you are prompted with the Message *Detected an Azure Function Project in folder*, you can safely close that message.

Create Custom Vision resources

Before you can train a model, you will need Azure resources for *training* and *prediction*. You can create **Custom Vision** resources for each of these tasks, or you can create a single **Azure AI Services** resource and use it for either (or both).

In this exercise, you'll create **Custom Vision** resources for training and prediction so that you can manage access and costs for these workloads separately.

1. In a new browser tab, open the Azure portal at <https://portal.azure.com>, and sign in using the Microsoft account associated with your Azure subscription.

Visual Studio Code

Editing evolved

Start

- New File...
- Open File...
- Open Folder...
- Clone Git Repository...
- Connect to...

Connect to remote development workspaces. (Ctrl+Alt+O)

You have no recent folders, [open a folder](#) to start.

Walkthroughs

- Learn the Fundamentals
- Boost your Productivity**
- Get Started with Python Development Updated
- Get Started with PowerShell Updated

[More...](#)

① Cloning git repository

"<https://github.com/MicrosoftLearning/mslearn-ai-vision>"...

Source: Git (Extension)

Cancer

☒ Show welcome page

Microsoft Azure



← user1-47587117@lodsprodmca.onmicrosoft.com

Enter password

.....

[Forgot my password](#)

Sign in

[Terms of use](#) [Privacy & cookies](#) ...



 Search

ENG
US

9:54
1/21/4 PM
2025

1



E

End >


•

Classify images with a Azure AI Vision custom model

2 Hr 55 Min Remaining

Instructions Resources Help   100%

► **Disclaimer** *Click to show or hide*

Tip: As you follow the instructions in this pane, whenever you see a  icon, you can use it to copy text from the instruction pane into the virtual machine interface. This is particularly useful to copy code; but bear in mind you may need to modify the pasted code to fix indent levels or formatting before running it!

Log into Windows as **Student** account with the password
T Pa55w.rd.

During the lab exercise, use the following credentials to sign into the Azure subscription that is provided for you:

- **User name:** User1-
47587117@LODSPRODMCA.onmicrosoft.com

- Password: T eJ0rSa!D2!

Create all Azure resources in the **ResourceGroup1** resource group.

⚠ Azure resources such as Virtual Machine names, sizes and locations must match what is specified in these steps. Due to the nature of Cloud Slice, deployments are limited to the scope defined in the instructional steps. If the resources provisioned differ from what is outlined in the instructions, a failure notification will be received, advising this was "disallowed by policy." If you believe you have received this notification in error, please submit a support request for our team to investigate further.

Classify Images with Azure AI Custom Vision

The **Azure AI Custom Vision** service enables you to create computer vision models that are trained on your own images. You can use it to train *image classification* and

13% Tasks Complete

End >

Note: If you already have an F0 custom vision service in your subscription, select **S0** for this one.

- ☐ 3. Wait for the resources to be created, and then view the deployment details and note that two Custom Vision resources are provisioned; one for training, and another for prediction (evident by the - **Prediction** suffix). You can view these by navigating to the resource group where you created them.

Important: Each resource has its own *endpoint* and *keys*, which are used to manage access from your code. To train an image classification model, your code must use the *training* resource (with its endpoint and key); and to use the trained model to predict image classes, your code must use the *prediction* resource (with its endpoint and key).

Marketplace - Microsoft Azure

https://portal.azure.com/#view/Microsoft_Azure_Marketplace/MarketplaceOffersBlade/searchQuery/custom%20vis...

Microsoft AzureSearch resources, services, and docs (G+/)CopilotUser1-47587117@LODS...LODS-PROD-MCA (LODSPROD...

Home > Create a resource >

Marketplace

Get Started

Service Providers

Management

Private Marketplace

Private Offer Management

My Marketplace

Favorites

My solutions

Recently created

Private plans

Categories

AI + Machine Learning (21)

IT & Management Tools (18)

Developer Tools (8)

Security (7)

Analytics (5)

Internet of Things (5)

custom vision

Pricing : AllOperating System : AllPublisher Type : AllProduct Type : All

Azure services only

Publisher name : All

Showing 1 to 20 of 56 results for 'custom vision'. Clear search

Tile view

Custom Vision
Microsoft
Azure Service
An AI service and end-to-end platform for applying computer vision to your specific scenario

Create

Trend Vision One™ Credits
Trend Micro
SaaS
Break down security silos with the power of a single cybersecurity platform.
Starts at \$1.05/1 year

Subscribe

Trend Vision One™
Trend Micro
SaaS
Break down security silos with the power of a single cybersecurity platform.
Starts at \$100,000.00/1 year

Subscribe

Trend Vision One for Microsoft Sentinel
Trend Micro
Azure Application
The Vision One connector provides the ability to ingest workbench alerts to Microsoft Sentinel.
Price varies

Create


Quest ThirdEye AI Vision Inspection
Quest Global
Azure Application
Quest ThirdEye AI Vision Inspection for Manufacturing

Joshua CyberRisk Vision
Almaviva SpA
Azure Application
Enabling actionable security data feed focused on NIS critical

AI-go Studio
OROBIX SRL
SaaS
AI-based vision inspection platform

WorldsNQ: The fastest and most accurate way to
Worlds
SaaS
A modern approach to computer vision, reducing time and labor by

Is Marketplace helpful?

 Loading your accounts...

Terms of Service

☒ I agree that my use of this service is governed by the [Microsoft Online Subscription Agreement](#), which incorporates the [Online Services Terms](#).

☒ I agree

End >





7. In the Custom Vision project, above the images, click **Train** to train a classification model using the tagged images. Select the **Quick Training** option, and then wait for the training iteration to complete (this may take a minute or so).
8. When the model iteration has been trained, review the *Precision*, *Recall*, and *AP* performance metrics - these measure the prediction accuracy of the classification model, and should all be high.

Note: The performance metrics are based on a probability threshold of 50% for each prediction (in other words, if the model calculates a 50% or higher probability that an image is of a particular class, then that class is predicted). You can adjust this at the top-left of the page.

Test the model

Now that you've trained the model, you can test it.

- ☐ 1. Above the performance metrics, click **Quick Test**.
- ☐ 2. In the **Image URL** box, type  `https://aka.ms/apple-image` and click 
- ☐ 3. View the predictions returned by your model - the probability score for *apple* should be the highest, like this:

Quick Test X

Image Lift

39% Tasks Complete

Microsoft Cognitive Services Custom Vision | Custom Vision: Classify Fruit - Performance

https://www.customvision.ai/projects/e3ec23a9-4b98-4daf-b106-80988ba02e8b#/performance

Classify Fruit

Training Images | Performance | Predictions

Train | Quick Test

Iterations

Probability Threshold: 50%

Iteration 1

Trained : 2 minutes ago with Food domain

✓ Publish | Prediction URL | Delete | Export

Iteration 1

Finished training on 1/21/2025, 10:08:39 PM using Food domain
Iteration id: 1b1360dd-49b3-4192-a62d-97be591f585a
Classification type: Multiclass (Single tag per image)

Precision ①

100.0%

Recall ①

100.0%

AP ①

100.0%

Performance Per Tag

Tag	Precision	Recall	A.P.	Image count
orange	100.0%	100.0%	100.0%	15
banana	100.0%	100.0%	100.0%	15

100% | Get started

The Custom Vision portal provides a convenient user interface that you can use to upload and tag images, and train models. However, in some scenarios you may want to automate model training by using the Custom Vision training API.

Note: In this exercise, you can choose to use the API from either the **C#** or **Python** SDK. In the steps below, perform the actions appropriate for your preferred language.

1. In Visual Studio Code, in the **Explorer** pane, browse to the **07-custom-vision-image-classification** folder and expand the **C-Sharp** or **Python** folder depending on your language preference.
2. Right-click the **train-classifier** folder and open an integrated terminal. Then install the Custom Vision Training package by running the appropriate command for your language preference:

C#

```
T dotnet add package Microsoft.Azure.Cogni
```

Python

```
T pip install azure-cognitiveservices-vision-computervision
```

- 3. View the contents of the **train-classifier** folder, and note that it contains a file for configuration settings:
 - **C#:** appsettings.json
 - **Python:** .env

Open the configuration file and update the configuration values it contains to reflect the endpoint and key for your Custom Vision *training* resource, and the project ID for the classification project you created previously. Save your changes.

The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays the project structure for 'MSLEARN-AI-VISION'. The 'train-classifier.py' file is selected. On the right, a PowerShell terminal window is open, showing a list of training tasks (apple, banana, Training ...) and a final status message 'Model trained!'. The terminal prompt is 'PS C:\Users\Student\Desktop\mslearn-ai-vision\Labfiles\07-custom-vision-image-classification\Python\train-classifier>'.

Classify images with a Azure AI Vision custom model

2 Hr 30 Min Remaining

Instructions Resources Help 100%

then used with the project ID to create a **Project** reference to your project.

- o The **Upload_Images** function retrieves the tags that are defined in the Custom Vision project and then uploads image files from correspondingly named folders to the project, assigning the appropriate tag ID.
- o The **Train_Model** function creates a new training iteration for the project and waits for training to complete.

- Return the integrated terminal for the **train-classifier** folder, and enter the following command to run the program:

C#

```
[T] dotnet run
```

Python

```
T python train-classifier.py
```

- ☒ 6. Wait for the program to end. Then return to your browser and view the **Training Images** page for your project in the Custom Vision portal (refreshing the browser if necessary).
- ☐ 7. Verify that some new tagged images have been added to the project. Then view the **Performance** page and verify that a new iteration has been created.

Publish the image classification model

Now you're ready to publish your trained model so that it can be used from a client application.

1. In the Custom Vision portal, on the **Performance** page, click ☐ **Publish** to publish the trained model with the following settings:
 - o **Model name:** fruit-classifier
 - o **Prediction Resource:** *The prediction resource you created previously which ends*

73% Tasks Complete

