Classify images with a Azure AI Vision custom model

2 Hr 9 Min Remaining

Instructions Resources Help  100%

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

Disclaimer

This lab content is presented as designed by Microsoft and has been adapted to Skillable’s platform for your use. If lab issues are identified that do not pertain directly to Skillable’s platform, please escalate these to your instructor for proper escalation.

**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 Pa55w.rd.

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

* **User name**: User1-46245525@LODSPRODMCA.onmicrosoft.com
* **Password**: a!JmaJU22!

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 an Azure AI Vision custom model

Azure AI Vision enables you to train custom models to classify and detect objects with labels you specify. In this lab, we'll build a custom image classification model to classify images of fruit.

Clone the repository for this course

If you have not already cloned the **Azure 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/mslearn-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.

Provision Azure resources

If you don't already have one in your subscription, you'll need to provision an **Azure AI Services** resource.

1. Open the Azure portal at https://portal.azure.com, and sign in using the Microsoft account associated with your Azure subscription.
2. In the top search bar, search for *Azure AI services*, select **Azure AI Services**, and create an Azure AI services multi-service account resource with the following settings:
   * **Subscription**: *Your Azure subscription*
   * **Resource group**: *Choose or create a resource group (if you are using a restricted subscription, you may not have permission to create a new resource group - use the one provided)*
   * **Region**: *Choose from East US, West Europe, West US 2\**
   * **Name**: *Enter a unique name*
   * **Pricing tier**: Standard S0

\*Azure AI Vision 4.0 custom model tags are currently only available in these regions.

1. Select the required checkboxes and create the resource.

We also need a storage account to store the training images.

1. In Azure portal, search for and select **Storage accounts**, and create a new storage account with the following settings:
   * **Subscription**: *Your Azure subscription*
   * **Resource Group**: *Choose the same resource group you created your Azure AI Service resource in*
   * **Storage Account Name**: customclassifySUFFIX
     + *note: replace the SUFFIX token with your initials or another value to ensure the resource name is globally unique.*
   * **Region**: *Choose the same region you used for your Azure AI Service resource*
   * **Primary service**: Azure Blob Storage or Azure Data Lake Storage Gen 2
   * **Primary workload**: Other
   * **Performance**: Standard
   * **Redundancy**: Locally-redundant storage (LRS)
2. While your storage account is being created, go to Visual studio code, and expand the **Labfiles/02-image-classification** folder.
3. In that folder, select **replace.ps1** and review the code. You'll see that it replaces the name of your storage account for the placeholder in a JSON file (the COCO file) we use in a later step. Replace the placeholder *in the first line only* of the file with the name of your storage account. Save the file.
4. Right-click on the **02-image-classification** folder and open an Integrated Terminal. Run the following command.

powershell

./replace.ps1

1. You can review the COCO file to ensure your storage account name is there. Select **training-images/training\_labels.json** and view the first few entries. In the *absolute\_url* field, you should see something similar to *"https://myStorage.blob.core.windows.net/fruit/…*. If you don't see the change expected, make sure you updated only the first placeholder in the PowerShell script.
2. Close both the JSON and PowerShell file, and go back to your browser window.
3. Your storage account should be complete. Go to your storage account.
4. Enable public access on the storage account. In the left pane, navigate to **Configuration** in the **Settings** group, and enable *Allow Blob anonymous access*. Select **Save**
5. In the left pane, in **Data storage**, select **Containers** and create a new container named fruit, and set **Anonymous access level** to *Container (anonymous read access for containers and blobs)*.

**Note**: If the **Anonymous access level** is disabled, refresh the browser page.

1. Navigate to fruit, select **Upload**, and upload the images (and the one JSON file) in **Labfiles/02-image-classification/training-images** to that container.

Create a custom model training project

Next, you will create a new training project for custom image classification in Vision Studio.

1. In the web browser, navigate to https://portal.vision.cognitive.azure.com/ and sign in with the Microsoft account where you created your Azure AI resource.
2. Select the **Customize models with images** tile (can be found in the **Image analysis** tab if it isn't showing in your default view).
3. Select the Azure AI Services account you created.
4. In your project, select **Add new dataset** on the top. Configure with the following settings:
   * **Dataset name**: training\_images
   * **Model type**: Image classification
   * **Select Azure blob storage container**: Select **Select Container**
     + **Subscription**: *Your Azure subscription*
     + **Storage account**: *The storage account you created*
     + **Blob container**: fruit
   * Select the box to "Allow Vision Studio to read and write to your blob storage"
5. Select the **training\_images** dataset.

At this point in project creation, you would usually select **Create Azure ML Data Labeling Project** and label your images, which generates a COCO file. You are encouraged to try this if you have time, but for the purposes of this lab we've already labeled the images for you and supplied the resulting COCO file.

1. Select **Add COCO file**
2. In the dropdown, select **Import COCO file from a Blob Container**
3. Since you have already connected your container named fruit, Vision Studio searches that for a COCO file. Select **training\_labels.json** from the dropdown, and add the COCO file.
4. Navigate to **Custom models** on the left, and select **Train a new model**. Use the following settings:
   * **Name of model**: classifyfruit
   * **Model type**: Image classification
   * **Choose training dataset**: training\_images
   * Leave the rest as default, and select **Train model**

Training can take some time - default budget is up to an hour, however for this small dataset it is usually much quicker than that. Select the **Refresh** button every couple minutes until the status of the job is *Succeeded*. Select the model.

Here you can view the performance of the training job. Review the precision and accuracy of the trained model.

Test your custom model

Your model has been trained and is ready to test.

1. On the top of the page for your custom model, select **Try it out**.
2. Select the **classifyfruit** model from the dropdown specifying the model you want to use, and browse to the **02-image-classification\test-images** folder.
3. Select each image and view the results. Select the **JSON** tab in the results box to examine the full JSON response.

Clean up resources

If you're not using the Azure resources created in this lab for other training modules, you can delete them to avoid incurring further charges.

1. Open the Azure portal at https://portal.azure.com, and in the top search bar, search for the resources you created in this lab.
2. On the resource page, select **Delete** and follow the instructions to delete the resource. Alternatively, you can delete the entire resource group to clean up all resources at the same time.