Transparency Note: Custom Vision
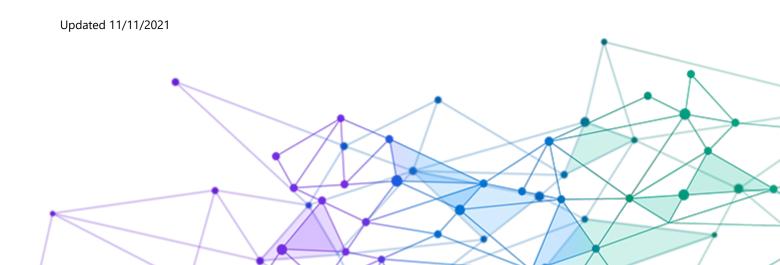
Updated 11/11/2021

Table of Contents

What is a Transparency Note?

An AI system includes not only the technology, but also the people who will use it, the people who will be affected by it, and the environment in which it is deployed. Creating a system that is fit for its intended purpose requires an understanding of how the technology works, what its capabilities and limitations are, and how to achieve the best performance. Microsoft's Transparency Notes are intended to help you understand how our AI technology works, the choices system owners can make that influence system performance and behavior, and the importance of thinking about the whole system, including the technology, the people, and the environment. You can use Transparency Notes when developing or deploying your own system, or share them with the people who will use or be affected by your system.

Microsoft's Transparency Notes are part of a broader effort at Microsoft to put our AI Principles into practice. To find out more, see the Microsoft AI principles.

Introduction to Custom Vision

Custom Vision is an image recognition service that lets you build, deploy, and improve your own image identification system. The service classifies images or detects objects in images based on their visual characteristics, allowing you to specify the categories to predict and example images for the system to learn from.

What Custom Vision does

The Custom Vision service uses machine learning to analyze images. You submit images that both include and lack the characteristics in question. You label the images yourself. Then, the algorithm trains itself using this data and calculates its own accuracy by testing itself on those same images. Once you've trained the algorithm, you can test, retrain, and eventually use it in your image recognition application or solution to infer predictions on new images. You can also export the model itself for offline use.

For more information, see What is Custom Vision?.

Terms and definitions

Term	Definition
Base model	Base models are a combination of model architecture and training data used to tune that architecture for a specific type of task (for example, image classification or object detection). Base models are built by Microsoft and are used as a starting point for the transfer learning process for various domains such as General, Food, Landmarks, Retail, Logos, and Products on Shelves.
Model training	This refers to the process of training a model based on customer-provided labeled images.
Transfer learning	The use of customer-provided training data to retrain a base model to address the specific problem the customer is trying to solve. The training data might be a combination of the classes they want to recognize and the type of images.
Class	A trained model has a set of classes it will assign when analyzing an input image. When you're building a model with Custom Vision, you define the set of classes you want the model to output, and provide labeled training data for each of the classes.
Class-accuracy pairs	The class-accuracy pair is a set of two values consisting of the name of the class and a float value for the confidence score associated with the class detection. For example, a class might be a type of fruit like a pineapple or pear represented in the image set. The class-accuracy pair is the specific class and the confidence score that the class is present in the image (e.g., Pineapple: 93.53%).
Image classification feature	This feature takes an image as an input, and outputs a set of class-accuracy pairs that are image-level properties (they don't specify a location within the image). For example, a pair might be fruit versus non-fruit, where the majority of the image is occupied by the respective class.
Object detection feature	This feature takes an image as an input, and outputs a set of class-accuracy pairs that also includes the bounding box coordinates for where in the image those classes were detected. For example, the model might detect the bounding box coordinates for where a vehicle is found in the image.
Bounding box	A set of four numerical values representing the x,y pixel coordinates of the top left corner of the detected object relative to the top left corner of the image, the width of the detected object.
Project	A logical grouping that encompasses the training data, the models trained from that data, and the prediction endpoints created for those models.

Functions of Custom Vision

Custom Vision's functionality can be divided into two features. **Image classification** applies one or more labels to an image. **Object detection** returns the coordinates in the image where the applied label(s) can be found for detected objects. Both features are delivered through APIs, SDKs and a website: https://customvision.ai.

Custom Vision supports the creation and use of custom computer vision models through the following high-level functions. They represent the two core activities you will complete to prepare your model for use:

- **Data labeling** is the process of annotating the training images with the classes of images that the model needs to classify. In the case of object detection, you annotate the training images with bounding boxes that surround the object to be detected in the image. Custom Vision provides you with a web portal where you can label training images with classes you select. Once the training data is labeled, you can use it for training the model.

- **Model training** uses the base model and transfer learning to train a model that's optimized for customer-provided images and corresponding classes. The quality of the model is highly dependent on the volume and quality of training data you provide. Custom Vision provides accuracy metrics, to approximate model performance based on a split of the training data provided.

When you're ready to use your model, you can make a *model prediction* by sending an image for processing. You can either host the model in the Custom Service cloud or you can export the model in a variety of formats to use as you want to.

To improve the quality of your model over time, you can sample data from your production deployment, or collect more data and retrain the model when that data is labeled. To assist with this process, you can use smart labeling, which suggests labels for a set of images that you upload.

Example use cases

You might use Azure Custom Vision for the following scenarios:

- **Automated visual alerts:** The ability to monitor a video stream and have alerts triggered when certain circumstances are detected. For example, you might want an alert when there is steam detected, or foam on a river, or an animal is present.
- **Improved efficiency of manual inspection:** In retail, product recognition enables you to reduce the time you or associates spend counting unique SKUs, or identifying whether all SKUs that should be on a shelf are present.
- **Expansion of inspection coverage:** When detecting defects, it's not always possible for a human to review all items coming off a manufacturing line. Instead, you can use Custom Vision to cover the set of items you aren't able to inspect manually, as well as inform which items you do inspect manually.
- **Improve object discoverability:** Labelling your images with metadata can make them easier to find later. For example, you might tag the images based on your product catalog or other visual features that you're interested in filtering on. Custom Vision allows you to label images with metadata at the time of ingestion.

Considerations when choosing a use case

We encourage customers to leverage Custom Vision in their innovative solutions or applications. However, here are some considerations when choosing a use case:

- **Not suitable for facial detection or recognition**. Custom Vision was not designed or tested to recognize or identify individuals in images. Instead, consider using Cognitive Services Face which has multiple face detectors available for use.
- **Not suitable for biometric identification**. Custom Vision was not designed or tested to verify the identity of individuals based on biometric markers such as iris recognition, fingerprint identification, or passports or other forms of ID for the purpose of identification and verification of a person.
- **Not suitable for training custom models for large-scale sets of images that contain hundreds of classes and tags**. Microsoft Azure Computer Vision has these capabilities as pre-built models for large scale image processing with thousands of tags.

- **Not suitable for detecting or extracting text**. Custom Vision was not designed or tested for processing text within images. Use Optical Character Recognition (OCR) for this purpose instead.
- **Not suitable for generating human-readable descriptions of images that can be used as alt-text for accessibility purposes**. Custom Vision was not designed or tested to generate descriptions for this purpose. Microsoft Azure Computer Vision has these capabilities to generate image descriptions and it's most suitable for this purpose.
- **Do not use Custom Vision for medical diagnosis** including for use as a medical device, clinical support, diagnostic tool, or other technology intended to be used in the diagnosis, cure, mitigation, treatment, or prevention of disease or other conditions, and no license or right is granted by Microsoft to use this capability for such purposes. This capability is not designed or intended to be implemented or deployed as a substitute for professional medical advice or healthcare opinion, diagnosis, treatment, or the clinical judgment of a healthcare professional, and should not be used as such. The customer is solely responsible for any use of Custom Vision for medical diagnosis.

Characteristics and limitations of Custom Vision

The quality of your classifier or object detector model built with Azure Custom Vision depends on the amount, quality, and variety of the labeled data you provide when training the model. The quality also depends on how balanced the overall dataset is between classes.

After you've trained your model, you can see the estimate of the project's performance customvision.ai. Custom Vision uses the images that you submitted for training to estimate precision, recall, and mean average precision. These three measurements of an image classifier's effectiveness are defined as follows:

- *Precision* is the percentage of identified classifications that were correct. For example, if the model identified 100 images as dogs, and 99 of them were actually of dogs, then the precision is 99 percent.
- *Recall* is the percentage of actual classifications that were correctly identified. For example, if there were actually 100 images of apples, and the model identified 80 as apples, the recall is 80 percent.
- *Mean average precision* (mAP) is the average value of the average precision (AP). AP is the area under the precision/recall curve (precision plotted against recall for each prediction made).

Probability threshold is the desired level of confidence that a prediction needs to have in order to be considered correct. When you interpret prediction calls with a high probability threshold, they tend to return results with high precision at the expense of recall. That is, the detected classifications are correct, but many remain undetected. A low probability threshold does the opposite: most of the actual classifications are detected, but there are more false positives within that set. With this in mind, you should set the probability threshold according to the specific needs of your project. By default, the probability threshold is 50% and can be set between 0% and 100%. To adjust the probability threshold go to customvision.ai on the **Performance** tab, find the **Probability Threshold** slider, and adjust it to your needs.

Evaluating and integrating Custom Vision for your use

Best practices to improve model accuracy

The process of building a Custom Vision model is iterative. Each time you train your model, you create a new iteration with its own, updated performance metrics. You can view all of your iterations in the left pane of the **Performance** tab for your project at customvision.ai.

A model will learn to make predictions based on arbitrary characteristics that your images have in common. To prevent overfitting, see How to improve your Custom Vision model.

We suggest that you test the model for an iteration with additional data. You can decide which iteration of the model to publish or export, and to use for inference.

Based on the model's performance, you need to decide if the model is appropriate for your use case and business needs. Here's an approach that you might take. You can deploy a Custom Vision model in an isolated environment, test the performance of the model relative to your use case, and then use the predictions to further train the model until it reaches the level of performance you want. For more information, see Quickstart: Build a classifier with the Custom Vision website, and Quickstart: Build an object detector with the Custom Vision website.

Learn more about responsible AI

[Microsoft AI principles](#)

[Microsoft responsible AI resources](#)

[Microsoft Azure Learning courses on responsible AI](#)

Learn more about Custom Vision

[What is Custom Vision?](#)

Contact us

[Give us feedback on this document](#)

About this document

© 2021 Microsoft Corporation. All rights reserved. This document is provided "as-is" and for informational purposes only. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it. Some examples are for illustration only and are fictitious. No real association is intended or inferred.

Published: 11/11/2021

Last updated: 11/11/2021