**Computer Vision and Its Applications**

This section introduces you to common concepts in computer vision (CV), and explains how you can use AWS DeepLens to start learning with computer vision projects. By the end of this section, you will be able to explain how to create, train, deploy, and evaluate a trash-sorting project that uses AWS DeepLens.

**Introduction to Computer Vision**

**Summary**

Computer vision got its start in the 1960s in academia. Since its inception, it has been an interdisciplinary field. Machine learning practitioners use computers to understand and automate tasks associated with the visual word.

Modern-day applications of computer vision use neural networks. These networks can quickly be trained on millions of images and produce highly accurate predictions.

Since 2010, there has been exponential growth in the field of computer vision. You can start with simple tasks like image classification and objection detection and then scale all the way up to the nearly real-time video analysis required for self-driving cars to work at scale.

In the video, you have learned:

* How computer vision got started
  + Early applications of computer vision needed hand-annotated images to successfully train a model.
  + These early applications had limited applications because of the human labor required to annotate images.
* Three main components of neural networks
  + *Input Layer*: This layer receives data during training and when inference is performed after the model has been trained.
  + *Hidden Layer*: This layer finds important features in the input data that have predictive power based on the labels provided during training.
  + *Output Layer*: This layer generates the output or prediction of your model.
* Modern computer vision
  + Modern-day applications of computer vision use neural networks call convolutional neural networks or CNNs.
  + In these neural networks, the hidden layers are used to extract different information about images. We call this process *feature extraction*.
  + These models can be trained much faster on millions of images and generate a better prediction than earlier models.
* How this growth occured
  + Since 2010, we have seen a rapid decrease in the computational costs required to train the complex neural networks used in computer vision.
  + Larger and larger pre-labeled datasets have become generally available. This has decreased the time required to collect the data needed to train many models.

**Computer Vision Applications**

**Summary**

Computer vision (CV) has many real-world applications. In this video, we cover examples of image classification, object detection, semantic segmentation, and activity recognition. Here's a brief summary of what you learn about each topic in the video:

* ***Image classification*** is the most common application of computer vision in use today. Image classification can be used to answer questions like *What's in this image?* This type of task has applications in *text detection* or *optical character recognition (OCR)* and *content moderation*.
* ***Object detection*** is closely related to image classification, but it allows users to gather more granular detail about an image. For example, rather than just knowing whether an object is present in an image, a user might want to know if there are *multiple instances of the same object* present in an image, or if *objects from different classes* appear in the same image.
* ***Semantic segmentation*** is another common application of computer vision that takes a pixel-by-pixel approach. Instead of just identifying whether an object is present or not, it tries to identify down the pixel level which part of the image is part of the object.
* ***Activity recognition*** is an application of computer vision that is based around videos rather than just images. Video has the added dimension of time and, therefore, models are able to detect changes that occur over time.

**New Terms**

* **Input Layer**: The first layer in a neural network. This layer receives all data that passes through the neural network.
* **Hidden Layer**: A layer that occurs between the output and input layers. Hidden layers are tailored to a specific task.
* **Output Layer**: The last layer in a neural network. This layer is where the predictions are generated based on the information captured in the hidden layers.

**Additional Reading**

* You can use the [AWS DeepLens Recipes](https://www.awsdeeplens.recipes/?utm_source=Udacity&utm_medium=Webpage&utm_campaign=Udacity%20AWS%20ML%20Foundations%20Course) website to find different learning paths based on your level of expertise. For example, you can choose either a *student* or *teacher* path. Additionally, you can choose between beginner, intermediate, and advanced projects which have been created and vetted by the AWS DeepLens team.
* You can check out the [AWS machine learning blog](https://aws.amazon.com/blogs/machine-learning/?utm_source=Udacity&utm_medium=Webpage&utm_campaign=Udacity%20AWS%20ML%20Foundations%20Course) to learn about recent advancements in machine learning. Additionally, you can use the [AWS DeepLens tag](https://aws.amazon.com/blogs/machine-learning/category/artificial-intelligence/aws-deeplens/?utm_source=Udacity&utm_medium=Webpage&utm_campaign=Udacity%20AWS%20ML%20Foundations%20Course) to see projects which have been created by the AWS DeepLens team.
* Ready to get started? Check out the [Getting started guide](https://docs.aws.amazon.com/deeplens/latest/dg/deeplens-getting-started.html?utm_source=Udacity&utm_medium=Webpage&utm_campaign=Udacity%20AWS%20ML%20Foundations%20Course) in the *AWS DeepLens Developer Guide*.