3.3 Convolutional neural network（CNN）

Deep learning relies on a convolutional neural network (CNN) model to transform an image into a predictive classification. CNN is a type of artificial neural network that uses convolutional layers to filter the input of useful information. It is the preferred network for image applications.

1. Artificial Neural Network

An artificial neural network is a biologically inspired computational model that is patterned after a network of neurons present in the human brain. At each level, the network transforms the input data by applying a nonlinear function to the weighted sum of the inputs. The intermediate output of one layer (called a feature) is used as the input to the next layer. The neural network learns multiple layers of nonlinear features (such as edges and shapes) through repeated transformations and then combines them in the last layer to create predictions (more complex objects).
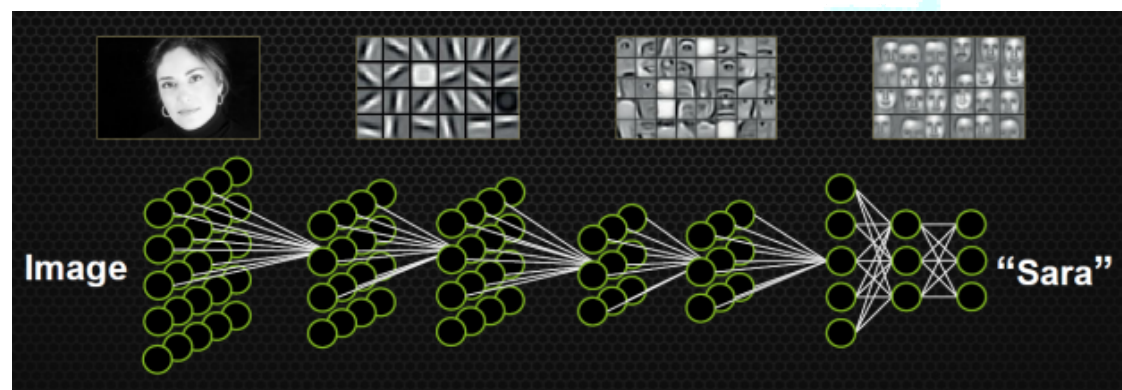
2. Convolution operation

A CNN-specific convolution operation combines input data (feature mapping) from one layer with a convolution kernel (filter) to form a transformation feature map of the next layer.

CNNs for image classification typically consist of an input layer (image), a series of hidden layers (convolution) for feature extraction, and a fully connected output layer (classification).
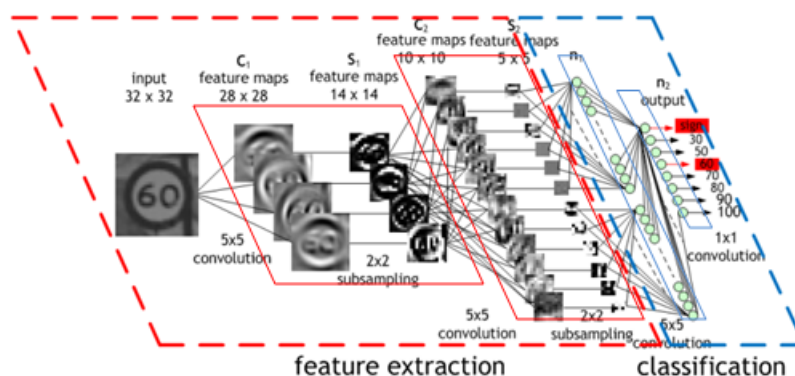
The input image of the traffic sign shown above is filtered by four 5x5

convolution kernels that create four feature maps that are subsampled through the largest pool.

The next layer applies 10 5x5 convolution kernels to these subsampled images and reassembles the feature maps. The last layer is a fully connected layer in which all generated features are combined and used in the classifier (basically logistic regression). Image courtesy of Maurice Peemen.

3. Use GPU to accelerate CNN

With a well-trained CNN model, the amount of computation required to train a CNN model and run reasoning can be very large, requiring a lot of computing resources and time.

Deep learning frameworks such as Caffe, TensorFlow, and PyTorch are optimized to run faster on the GPU. The framework takes advantage of the GPU's parallel processing capabilities, if any, to speed up training and reasoning tasks.
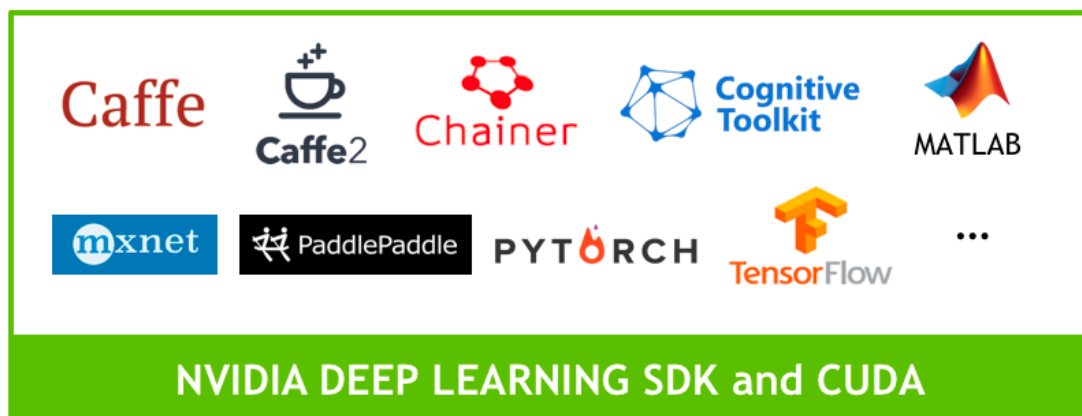
The Jetbot robot car uses the Jetson Nano development board as the main controller, including a 128-core NVIDIA Maxwell GPU. Because it can run a complete training framework, it can also retrain the network through transfer learning. In the following lessons, several pre-trained neural network models will be displayed using the Pytorch framework. You can also retrain new nerves yourself. The network model is applied in other different scenarios.