

Using Deep Learning in Machine Vision

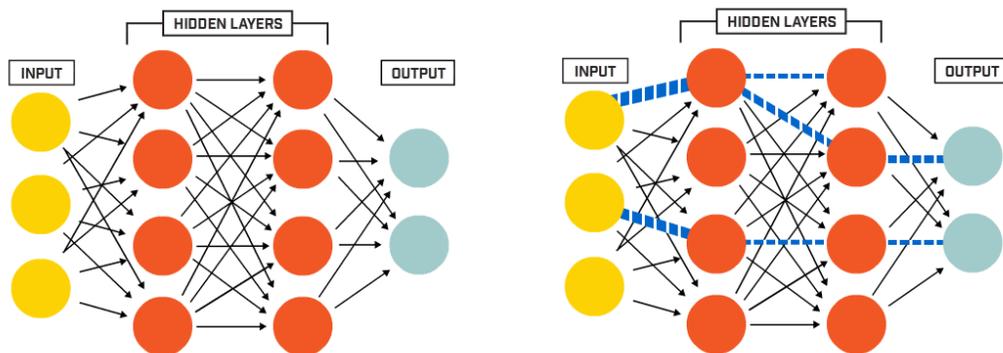
What's inside:

- What is deep learning
- Why is deep learning taking off now?
- Multiple applications
- How to implement a system

More and more, machine vision systems are expected to make dynamic, automated decisions based on variable conditions. The amount of time and effort to develop these systems can be daunting. Today, the advent of deep learning is changing this landscape and putting automation within the reach of many. Resources such as open-source libraries, Nvidia hardware, and FLIR cameras are helping to make this change happen. FLIR cameras have advanced features that minimize the image pre-processing required for neural network training, work seamlessly with platforms such as NVidia Jetson TX-2 and Drive PX 2, and offer 24/7 reliability for trouble-free deployment.

What is deep learning?

Deep learning is a form of machine learning that uses neural networks with many “deep” layers between the input and output nodes. By training a network on a large data set, a model is created that can be used to make accurate predictions based on input data. In neural networks used for deep learning, each layer’s output is fed forward to the input of the next layer. The model is optimized iteratively by changing the weights of the connections between layers. On each cycle, feedback on the accuracy of the model’s predictions is used to guide changes in the connection weighting.



*A neural network containing “deep” hidden layers between the input and output.
Changes in relative weights of inputs.*

Deep learning is transforming industries everywhere by automating processes that were too complex for traditional vision applications. Easy to use frameworks, affordable, accelerated Graphics Processing Unit (GPU) hardware, and cloud computing platforms have made deep learning accessible to everyone.

Cucumber sorting example

An example of deep learning's accessibility is found in the story of a Japanese farmer who configured Google's TensorFlow framework and Cloud ML to grade his cucumbers. Using TensorFlow, he trained a neural network with sample images for each cucumber grade. The system learned to distinguish cucumber grades based on features in the sample images. As cucumbers move through the farmer's sorting machine, they are imaged. The trained neural network then classifies the cucumbers and directs the sorting machine to divert them into the correct bins.

Why is Deep Learning taking off now?

GPU accelerated hardware: more power, less cost

The architecture of GPUs, which uses a large number of processors to perform a set of coordinated computations in parallel (known as a "massively parallel" architecture), is ideal for deep learning systems. Ongoing development from Nvidia has resulted in large increases in the power, efficiency, and affordability of GPU-accelerated computing platforms. This technology is available in a range of form factors such as compact embedded systems based on the Jetson TX1 and TX2, PC GPUs like the GTX 1080, and dedicated AI platforms like the Nvidia DGX-1 and Drive PX 2.

Democratization of deep learning frameworks

In addition to the development of easy-to-use frameworks, the widespread availability of tutorials and online courses has contributed to deep learning accessibility. C++ wrappers, including Google's TensorFlow and the open source packages Caffe, Torch, and Theano, enable users to quickly build and train their own Deep Neural Networks (DNNs). The general purpose TensorFlow is a great starting point, while Caffe's GPU optimization makes it an excellent choice for deployment on the Jetson TX1 and TX2.

The Nvidia CUDA Deep Neural Network (cuDNN) library provides developers with highly-optimized implementations of common deep learning functions, further streamlining development for these platforms.

Better prices, shorter lead times

The availability of discrete, off-the-shelf cameras and embedded platforms gives traffic system designers the flexibility to tailor systems to fit their projects. Separate cameras and processing hardware enable a simple, independent upgrade path for each component. This ecosystem results in better prices and shorter lead times versus dedicated smart cameras.

Multiple applications

While the development of autonomous vehicles attracts a lot of media attention, deep learning has many other applications. Deep learning can solve a wide range of problems, from helping doctors to more accurately interpret CT scans to automatic text translation and traffic flow optimization across cities. Deep learning is a powerful tool for designers of automated optical inspection systems (AOI). By learning from parts that are known to be good, deep learning powered AOI software like ViDi Red can detect defects as well as learn to recognize acceptable variations.

Continued training of deep learning systems enables them to respond to changing conditions. A company named HERE is working to deploy their deep-learning-powered mapping system in autonomous vehicles. Their technology will generate continuously updated maps with a resolution of 10-20cm. Using deep learning, HERE's maps will include the precise locations of fixed objects like signage, and temporary driving hazards like construction work.

How to implement a system

Training data acquisition

Designers must train a deep learning model before deploying it. High-quality training data is essential to achieving accurate results. High-performance cameras provide the best possible training imagery to systems that make decisions based on visual input.

On-camera image processing simplifies the data normalization required prior to training. Camera features like precise control over auto-algorithms, sharpening, pixel format conversion, and FLIR's advanced debayering and Color Correction Matrix, optimize images. FLIR's strict quality control during manufacturing minimizes variation in camera performance, reducing the need for pre-training normalization.

For applications that image moving subjects, global shutter sensors read all pixels simultaneously, eliminating distortion caused by the subject moving during the readout process. Many FLIR machine vision cameras use Sony Pregius global shutter CMOS sensors. They have 72dB of dynamic range and less than 3e- read noise, enabling them to simultaneously capture details in brightly-lit and shaded areas, and providing excellent low-light performance.

Low light applications like night-time security and fluorescence microscopy benefit from the pixel structure of Back-Side-Illuminated (BSI) Sony Exmor R and Starvis sensors.

These devices trade readout speed for greater quantum efficiency, making them small, inexpensive sensors with great low-light performance.

Train on specialized hardware

Once enough training data has been collected, it's time to train your model. To expedite this process, it is possible to use a PC with one or more CUDA enabled GPUs or specialized AI training hardware like the Nvidia DGX-1. Cloud computing platforms that specialize in deep learning are also available.



GPU-accelerated training of deep learning systems is much faster than CPU

Deploy to an embedded system

Once the training of your deep learning model is complete, it's time to deploy it to the field. Compact and powerful GPU-accelerated embedded platforms enable applications where space and power requirements preclude a traditional PC, and limited internet connectivity necessitates on-the edge computing. These systems are based on ARM processor architecture and typically run on a Linux based OS.

Many industrial applications rely on systems with more than one camera. With FLIR machine vision cameras, system designers have the freedom to accurately trigger multiple cameras over GPIO or software. The IEEE 1588 Precision Time Protocol (PTP) enables camera clock synchronization to a common time base or a GPS time signal with no user oversight. The MTBF of multi-camera systems decreases with every additional camera, making highly reliable cameras critical to building robust systems. The design and testing of FLIR Machine vision cameras ensures 24/7 reliability, minimizing downtime and maintenance.

The Nvidia Jetson TX1 and TX2 are powerful and efficient GPU-accelerated embedded platforms that support USB 3.1 Gen 1 and GigE vision cameras. Specialized Jetson carrier boards provide I/O connectivity and application-specific features. The SmartCow TERA+ supports up to 8 GigE cameras natively with the use of a managed switch, and RS-232 and RS-485 serial communication. SmartCow also provides a Caffe wrapper which streamlines the design and deployment of deep learning powered vision applications on

the TERA+ hardware. The Connect Tech Cogswell Carrier supports USB 3.1 Gen 1 and Power Over Ethernet GigE cameras.

The Nvidia Drive PX 2 is an open automotive AI platform built around two Pascal GPU cores. Capable of eight TFLOPS, the Drive PX 2 has the equivalent computing power of 150 Macbook Pros. The drive PX 2 supports deep learning applications for autonomous vehicle guidance. In addition to USB 3.1 Gen 1 and GigE vision cameras, it has inputs for cameras using the automotive GMSL camera interface.

First steps with deep learning – easier than you think!

Dive into deep learning with these great tutorials:

<https://developer.nvidia.com/embedded/twodaystodemo>

<http://blog.udacity.com/2017/01/siraj-raval-deep-learning-nanodegree-foundation-program.html>

<http://www.flir.com/deeplearning>