

Deep Learning in OpenCV

Vadim Pisarevsky edited this page on 2 Aug 2017 · 4 revisions

Deep Learning is the most popular and the fastest growing area in Computer Vision nowadays. Since OpenCV 3.1 there is DNN module in the library that implements forward pass (inferencing) with deep networks, pre-trained using some popular deep learning frameworks, such as Caffe. In OpenCV 3.3 the module has been promoted from `opencv_contrib` repository to the main repository (<https://github.com/opencv/opencv/tree/master/modules/dnn>) and has been accelerated significantly.

The module has no any extra dependencies, except for `libprotobuf`, and `libprotobuf` is now included into OpenCV.

The supported frameworks:

- Caffe 1
- TensorFlow
- Torch/PyTorch

The supported layers:

- AbsVal
- AveragePooling
- BatchNormalization
- Concatenation
- Convolution (including dilated convolution)
- Crop
- Deconvolution, a.k.a. transposed convolution or full convolution
- DetectionOutput (SSD-specific layer)
- Dropout
- Eltwise (+, *, max)
- Flatten
- FullyConnected
- LRN
- LSTM
- MaxPooling
- MaxUnpooling
- MVN
- NormalizeBBBox (SSD-specific layer)
- Padding
- Permute
- Power
- PReLU (including ChannelPReLU with channel-specific slopes)
- PriorBox (SSD-specific layer)
- ReLU
- RNN
- Scale
- Shift
- Sigmoid

► Pages 87

- [Home](#)
 - [Changelog \(older\)](#)
 - New functionality discussion
 - [RGBD](#)
 - [Documentation improvement plan](#)
 - [Android](#)
 - [Release Notes](#)
 - [Building](#)
 - [CiteOpenCV](#)
 - [OpenCVLogo](#)
- [Deep Learning in OpenCV](#)
 - [DNN Efficiency](#)
 - [TensorFlow Object Detection API](#)
 - [Intel's Deep Learning Inference Engine backend](#)
- [OpenCV 3](#)
 - [OpenCL optimizations](#)
 - [CPU optimizations](#)
 - [Profiling OpenCV Applications](#)
 - [MediaSDK encode/decode backend](#)
 - [Building more compact applications with OpenCV](#)
- [Development process](#)
 - [How to contribute](#)
 - [Coding style guide](#)
 - [Evolution Proposals](#)
 - [Contributors](#)
 - [Tech guides:](#)
 - [Working with Git](#)
 - [Windows 7 guide](#)
 - [Unix based guide](#)
 - [Meeting notes](#)
 - [Template](#)
 - [2018](#)
 - [2017](#)
 - [2016](#)
 - [2015](#)
 - [2014](#)
 - [2013](#)
 - [2012](#)
 - [2011](#)
 - [2010](#)
 - [2009](#)
 - [2008](#)
 - [QA](#)

- Slice
- Softmax
- Split
- TanH

The module includes some SSE, AVX, AVX2 and NEON acceleration of the performance-critical layers. There is also constantly-improved Halide backend. OpenCL (libdnn-based) backend is being developed and should be integrated after OpenCV 3.3 release. Here you may find the up-to-date benchmarking results: [DNN Efficiency](#)

The following networks have been tested and known to work:

- AlexNet
- GoogLeNet v1 (also referred to as Inception-5h)
- ResNet-34/50/...
- SqueezeNet v1.1
- VGG-based FCN (semantical segmentation network)
- ENet (lightweight semantical segmentation network)
- VGG-based SSD (object detection network)
- MobileNet-based SSD (light-weight object detection network)

The provided API (for C++ and Python) is very easy to use, just load the network and run it. Multiple inputs/outputs are supported. Here are the examples: <https://github.com/opencv/opencv/tree/master/samples/dnn>.

There is Habrahabr article describing the module: <https://habrahabr.ru/company/intel/blog/333612/> (in Russian).

- [Android tests](#)
- [Using performance tests](#)
- [Writing performance tests](#)

- Tutorials
 - [CARMA platform](#)
 - [Debug in Visual Studio](#)
 - [Displaying multiple images](#)
 - [Face detection](#)
 - [POSIT](#)
 - [WindowsRT](#)
 - [ANCT 2013 \(ru\)](#)
- Computer Vision and Pattern Recognition
 - [2015](#)
 - [2014](#)
 - [2010](#)
- Google summer of code
 - [OpenCV GSoC Application](#)
 - [2018](#)
 - [2017](#)
 - [2016](#)
 - [2015](#)
 - [2014](#)
 - [2011 - image collage](#)
- Vision challenge
- Workshops
 - [OpenCV_Talks](#)

Clone this wiki locally

<https://github.com/opencv>



© Copyright 2018, OpenCV team