

Using Sample Applications

The Inference Engine sample applications are simple console applications that demonstrate how you can use the Intel's Deep Learning Inference Engine in your applications.

The Deep Learning Inference Engine release package provides the following sample applications available in the samples directory in the Inference Engine installation directory:

- [Image Classification Sample \(/node/317737f7-a00c-4517-bdd0-065952b22c67\)](/node/317737f7-a00c-4517-bdd0-065952b22c67) - Inference of image classification networks like AlexNet and GoogLeNet
- [Image Segmentation Sample \(/node/33c59b88-0e11-40b8-98e8-2b6108c6d323\)](/node/33c59b88-0e11-40b8-98e8-2b6108c6d323) - Inference of image segmentation networks like FCN8
- [Multiple Outputs Sample \(/node/7153bdb0-ff91-41d2-a2cf-6840d51af250\)](/node/7153bdb0-ff91-41d2-a2cf-6840d51af250) - Inference of networks with several outputs
- [Object Detection Sample \(/node/4b21ede0-4ddd-49fd-a518-af310ebab904\)](/node/4b21ede0-4ddd-49fd-a518-af310ebab904) - Inference of object detection networks like Faster R-CNN
- [Object Detection for SSD Sample \(/node/1c49114e-50d6-4537-91a2-95356986afdd\)](/node/1c49114e-50d6-4537-91a2-95356986afdd) - Inference of object detection networks like SSD-VGG
- [Object Detection for YOLO Sample \(/node/be1bcab1-e653-4f3b-aa1d-d77d8b231710\)](/node/be1bcab1-e653-4f3b-aa1d-d77d8b231710) - Inference of object detection networks like YOLO
- [Extensibility Sample \(/node/c07d7df8-a176-4534-a87a-61a426756325\)](/node/c07d7df8-a176-4534-a87a-61a426756325) - Inference of networks with custom layers
- [Neural Style Transfer Sample \(/node/2649a11f-7911-43dc-aa2a-c896d31e53a5\)](/node/2649a11f-7911-43dc-aa2a-c896d31e53a5) - Inference using models of style transfer topology
- [Validation Application \(/node/b0439442-2985-4a47-99bc-18d627cc9d0c\)](/node/b0439442-2985-4a47-99bc-18d627cc9d0c) - Infers pack of images resulting in total accuracy.

Building the Sample Applications on Linux* OS

The recommended Linux build environment is the following:

- Ubuntu* 14.04 LTS 64-bit, Ubuntu* 16.04 LTS 64-bit or CentOS* 7.3 64-bit
- GCC* version 4.8 or higher
- CMake* version 2.8 or higher
- OpenCV 3.3 (required for the Validation App only).

You can build the sample applications using the *CMake* file in the *samples* directory.

Create a new directory and change your current directory to the new one:

```
1 | $ mkdir build
2 | $ cd build
```

To build samples with debug information, use the following command:

```
1 | cmake -DCMAKE_BUILD_TYPE=Debug <path_to_inference_engine_samples_directory>
```

Run *Make* to build the application:

```
1 | make
```

For ease of reference, the Inference Engine installation folder is referred to as `IE_INSTALL`.

For convenience, you can set the following environment variables: `export IE_INSTALL="PATH_TO_YOUR_INSTALL_DIR"`

After that you can find binaries for all samples applications in the `$IE_INSTALL/deployment_tools/inference_engine/bin/intel64/Release` folder:

Building the Sample Applications on Microsoft Windows* OS

The recommended Windows build environment is the following:

- Microsoft Windows* 10
- Microsoft* Visual Studio* 2015
- CMake* version 2.8 or later
- OpenCV* 3.3
- Intel® C++ Compiler 2017 Redistributable package for Windows

Generate Microsoft Visual Studio solution file using `create_msvc_solution.bat` file in the `samples` directory and then build the solution `samples.sln` in the Microsoft Visual Studio 2015.

Running the Sample Applications

Before running compiled binary files, make sure your application can find the Inference Engine libraries.

To make the Inference Engine recognize required libraries availability, use the `setvars.sh` script, which will set all necessary environment variables.

For that, run (assuming that you are in a `$IE_INSTALL/deployment_tools/inference_engine/bin/intel64/Release` folder):

```
1 | source ../../setvars.sh
```

What is left is running the required sample with appropriate commands, providing IR information. Please note that Inference Engine assumes that weights are in the same folder as `.xml` file.

For more details about running each sample application, see the following topics:

- [Image Classification Sample \(/node/317737f7-a00c-4517-bdd0-065952b22c67\)](#)
- [Image Segmentation Sample \(/node/33c59b88-0e11-40b8-98e8-2b6108c6d323\)](#)
- [Multiple Outputs Sample \(/node/7153bdb0-ff91-41d2-a2cf-6840d51af250\)](#)

- [Object Detection Sample \(/node/4b21ede0-4ddd-49fd-a518-af310ebab904\)](/node/4b21ede0-4ddd-49fd-a518-af310ebab904)
- [Object Detection for SSD Sample \(/node/1c49114e-50d6-4537-91a2-95356986afdd\)](/node/1c49114e-50d6-4537-91a2-95356986afdd)
- [Object Detection for YOLO Sample \(/node/be1bcab1-e653-4f3b-aa1d-d77d8b231710\)](/node/be1bcab1-e653-4f3b-aa1d-d77d8b231710)
- [Extensibility Sample \(/node/c07d7df8-a176-4534-a87a-61a426756325\)](/node/c07d7df8-a176-4534-a87a-61a426756325)
- [Neural Style Transfer Sample \(/node/2649a11f-7911-43dc-aa2a-c896d31e53a5\)](/node/2649a11f-7911-43dc-aa2a-c896d31e53a5)
- [Validation Application \(/node/b0439442-2985-4a47-99bc-18d627cc9d0c\)](/node/b0439442-2985-4a47-99bc-18d627cc9d0c)

See Also

- [Introduction to Intel's Deep Learning Inference Engine \(/node/293a56b8-e0e9-4290-973a-303c142d9b10\)](/node/293a56b8-e0e9-4290-973a-303c142d9b10)

For more complete information about compiler optimizations, see our [Optimization Notice \(/en-us/articles/optimization-notice#opt-en\)](/en-us/articles/optimization-notice#opt-en).