20210228_intel_deeplearning_day_hitnet_demo
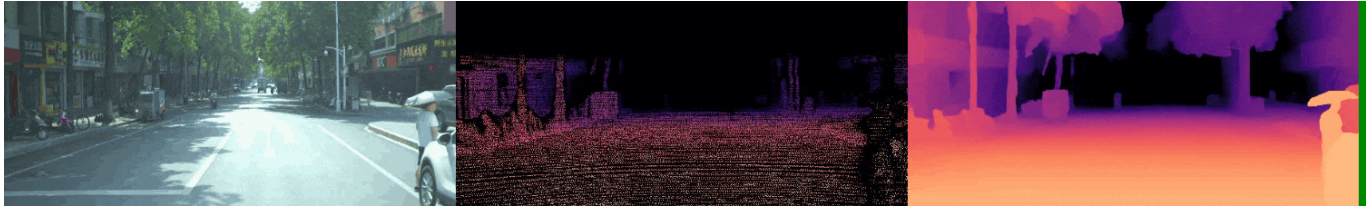
https://www.intel.co.jp/content/www/jp/ja/now/iot-planet/deep-learning-day.html

1. Overview / 概要

This is a demonstration of the steps to convert and infer HITNet, a stereo depth estimation model, using a custom build of OpenVINO.
OpenVINOをカスタムビルドしてステレオ深度推定モデルのHITNetを変換し、推論するまでの手順のデモです。

2. Environment / 環境

- Ubuntu 20.04 x86_64
- Docker 20.10.12, build e91ed57
- OpenVINO commit hash: e89db1c6de8eb551949330114d476a2a4be499ed
- ONNX

3. Overall flow / 全体の流れ

In order to optimize the process as much as possible, the following processing flow is adopted.
TensorFlow `pb` -> TensorFlow `saved_model` -> TensorFlow Lite `tflite` -> ONNX `onnx` -> OpenVINO IR `xml/bin`

- 4-1. Procurement of original model .pb / .pb オリジナルモデル.pbの調達
- 4-2. Convert .pb to saved_model / .pbをsaved_modelに変換
- 4-3. Convert saved_model to ONNX / saved_modelをONNXに変換
- 4-4. Building OpenVINO / OpenVINOのビルド
- 4-5. Convert ONNX to OpenVINO IR / ONNXをOpenVINO IRへ変換
- 4-6. Download the Dataset / Datasetのダウンロード
- 4-7. HITNet's ONNX demo / HITNetのONNXデモ
- 4-8. HITNet's OpenVINO demo / HITNetのOpenVINOデモ
- 4-9. HITNet's TensorRT demo / HITNetのTensorRTデモ

4. Procedure / 手順

4-1. Procurement of original model .pb / オリジナルモデル.pbの調達

Download the official HITNet model published by Google Research here. The file to be downloaded is a Protocol Buffers format file.
こちら のGoogle Researchが公開しているHITNet公式モデルをダウンロードします。ダウンロードするファイルはProtocol Buffers形式のファイルです。

```
$ git clone
https://github.com/PINTO0309/20210228_intel_deeplearning_day_hitnet_demo
$ cd 20210228_intel_deeplearning_day_hitnet_demo

# [1, ?, ?, 2], Grayscale image x2
$ wget https://storage.googleapis.com/tensorflow-
graphics/models/hitnet/default_models/eth3d.pb
or
# [1, ?, ?, 6], RGB image x2
$ wget https://storage.googleapis.com/tensorflow-
graphics/models/hitnet/default_models/flyingthings_finalpass_xl.pb
or
# [1, ?, ?, 6], RGB image x2
$ wget https://storage.googleapis.com/tensorflow-
graphics/models/hitnet/default_models/middlebury_d400.pb
```
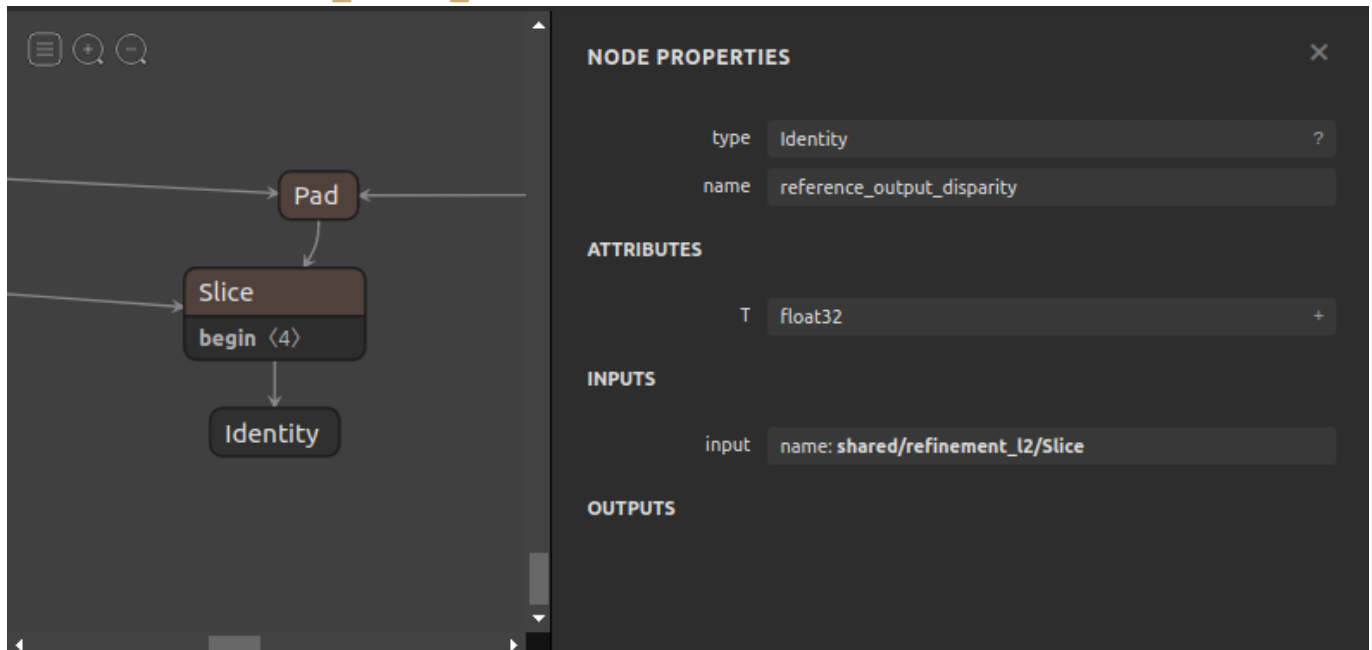
Use Netron to check the structure of the model. In the case of eth3d, two grayscale images of one channel are used as input. The name of the input is `input`.

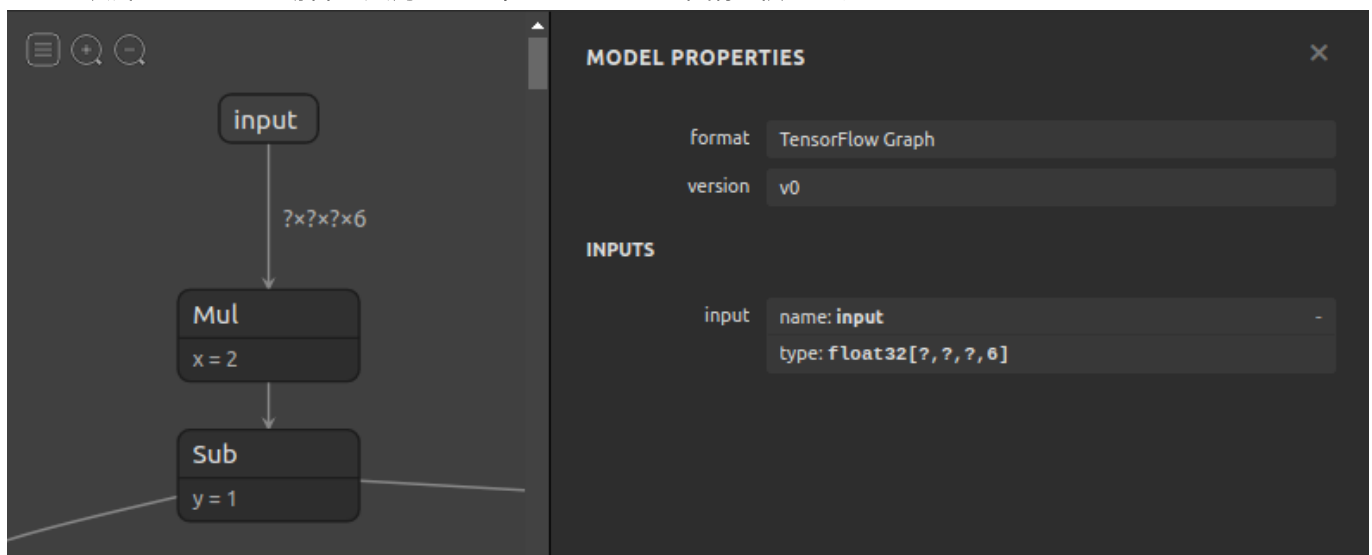モデルの構造を確認するには、Netron を使用します。eth3dの場合、1チャンネルのグレースケール画像2枚を入力として使用します。入力の名前は `input` です。
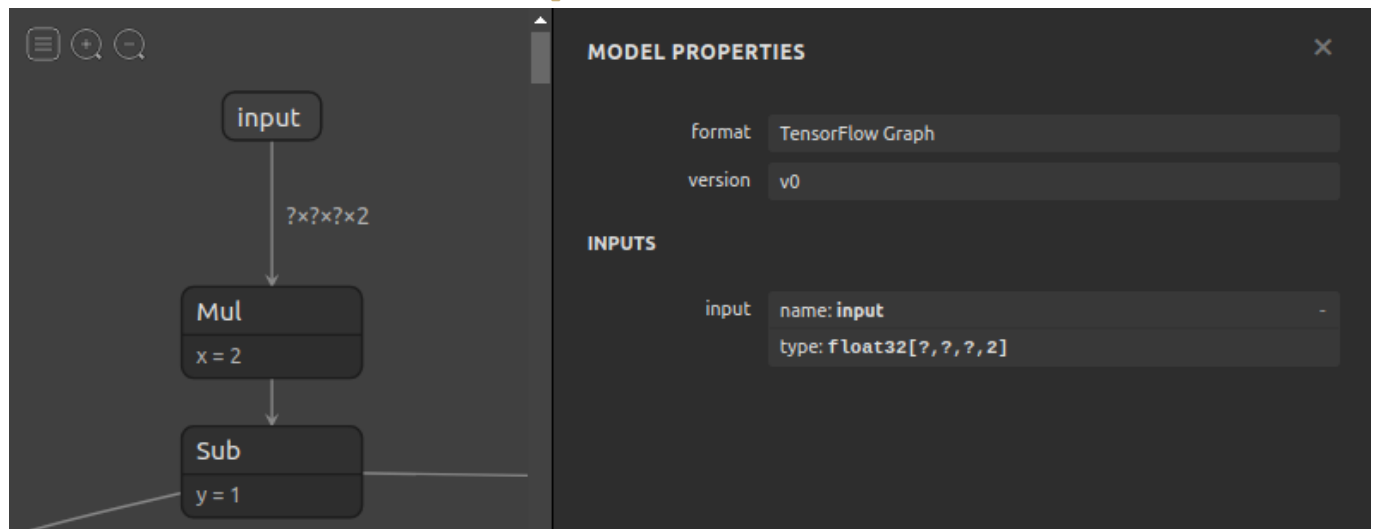
The name of the output is `reference_output_disparity`.

出力の名前は `reference_output_disparity` です。

For non-eth3d, the input is two 3-channel RGB images.
eth3d以外のモデルの場合、入力は3チャンネルのRGB画像2枚です。

↥ Back to top

4-2. Convert .pb to saved_model / .pbをsaved_modelに変換

Start a Docker container with all the latest versions of the various major frameworks such as OpenVINO, TensorFlow, PyTorch, ONNX, etc. Note that the Docker Image is quite large, 26GB, since all the huge frameworks such as CUDA and TensorRT are also installed. Also, in order to launch the demo with GUI from within the Docker container, many launch options are specified, such as `xhost`, `--gpus`, `-v`, `-e`, `--net`, `--privileged`, etc., but they do not need to be specified if you do not want to use the GUI. If you want to know what kind of framework is implemented in a Docker container, please click here.
OpenVINOやTensorFlowやPyTorchやONNXなどの各種主要フレームワークの最新バージョンが全て導入されたDockerコンテナを起動します。CUDAやTensorRTなどの巨大なフレームワークも全てインストールされているため、Docker Imageは26GBとかなり大きいことに注意してください。また、Dockerコンテナの中からGUIを使用したデモを起動するため、`xhost`, `--gpus`, `-v`, `-e`, `--net`, `--privileged` などの多くの起動オ

プションを指定していますが、GUIを使用しない場合は指定不要です。どのようなフレームワークが導入されたDockerコンテナかを知りたい場合は こちら をご覧ください。

```
$ xhost +local: && \
docker run --gpus all -it --rm \
-v `pwd`:/home/user/workdir \
-v /tmp/.X11-unix/:/tmp/.X11-unix:rw \
--net=host \
-e XDG_RUNTIME_DIR=$XDG_RUNTIME_DIR \
-e DISPLAY=$DISPLAY \
--privileged \
ghcr.io/pinto0309/openvino2tensorflow:latest
```

```
$ MODEL=eth3d
or
$ MODEL=flyingthings_finalpass_xl
or
$ MODEL=middlebury_d400

$ pb_to_saved_model \
--pb_file_path ${MODEL}.pb \
--inputs input:0 \
--outputs reference_output_disparity:0 \
--model_output_path ${MODEL}/saved_model
```

A sample without GUI is shown below.
GUIを使用しない場合のサンプルは下記のとおりです。

```
$ docker run -it --rm \
-v `pwd`:/home/user/workdir \
ghcr.io/pinto0309/openvino2tensorflow:latest
```

```
$ MODEL=eth3d
or
$ MODEL=flyingthings_finalpass_xl
or
$ MODEL=middlebury_d400

$ pb_to_saved_model \
--pb_file_path ${MODEL}.pb \
--inputs input:0 \
--outputs reference_output_disparity:0 \
--model_output_path ${MODEL}/saved_model
```

Let's check the shape of the generated `saved_model`, using the standard TensorFlow tool `saved_model_cli`.Of the input NHWC shape `batch,height,width,channel`, the height and width are undefined `-1`.
生成された `saved_model` の形状を確認してみます。TensorFlowの標準ツール `saved_model_cli` を使用します。入力のNHWC形状 **バッチサイズ,高さ,幅,チャンネル** のうち、高さと幅が未定義の `-1` となっています。

```
$ saved_model_cli show --dir middlebury_d400/saved_model/ --all

MetaGraphDef with tag-set: 'serve' contains the following SignatureDefs:

signature_def['serving_default']:
  The given SavedModel SignatureDef contains the following input(s):
    inputs['input'] tensor_info:
        dtype: DT_FLOAT
        shape: (-1, -1, -1, 6)
        name: input:0
  The given SavedModel SignatureDef contains the following output(s):
    outputs['reference_output_disparity'] tensor_info:
        dtype: DT_FLOAT
        shape: (-1, -1, -1, -1)
        name: reference_output_disparity:0
  Method name is: tensorflow/serving/predict
```

↥ Back to top

4-3. Convert saved_model to ONNX / saved_modelをONNXに変換

The tool `saved_model_to_tflite` introduced in the Dokcer container is used to generate `tflite` from `saved_model`. The tool `tensorflow-onnx` can be used to generate `onnx` from `saved_model` immediately, but I will convert it once to `tflite` to make it as optimized as possible. The `--input_shapes` option can be used to fix undefined input shapes to a specified size.
Dokcerコンテナに導入されている `saved_model_to_tflite` というツールを使用して `saved_model` から `tflite` を生成します。 公式の `tensorflow-onnx` というツールを使用すると `saved_model` から即座に `onnx` を生成することが可能ですが、なるべく最適化を行うためにあえて一度 `tflite` へ変換します。 `--input_shapes` オプションを使用することで未定義の入力形状を指定のサイズへ固定することができます。

```
$ H=480
$ W=640
$ saved_model_to_tflite \
--saved_model_dir_path ${MODEL}/saved_model \
--input_shapes [1,${H},${W},6] \
--model_output_dir_path ${MODEL}/saved_model_${H}x${W} \
--output_no_quant_float32_tflite
```

Check the input and output structure of the generated TFLite. At this point, TensorFlowLite's optimizer has already removed a large number of unnecessary operations or merged multiple operations into a clean and

simple structure.

生成されたTFLiteの入力と出力の構造を確認します。この時点ですでにTensorFlowLiteのオプティマイザによって不要なオペレーションが大量に削除されたり、あるいは複数のオペレーションが融合して綺麗でシンプルな構造に変換されています。
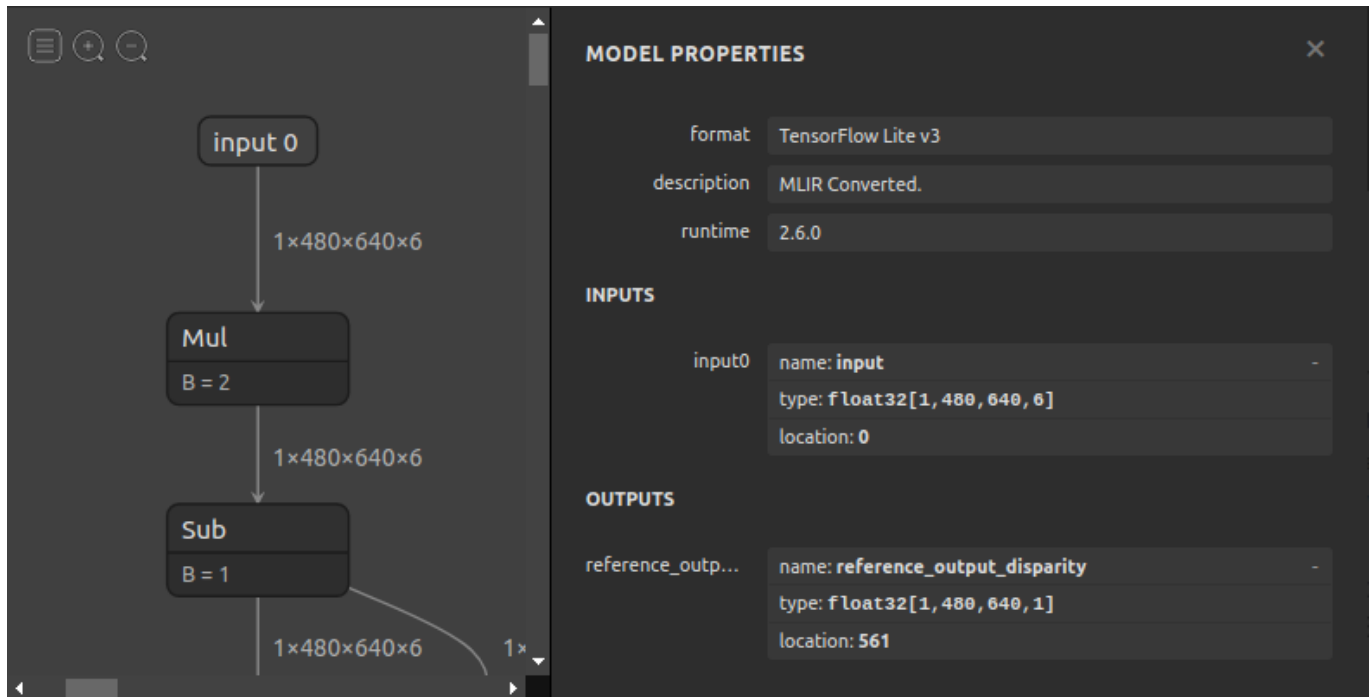

Next, convert `tflite` to `onnx`. I will use `tensorflow-onnx` here. `--inputs-as-nchw input` is an option to convert the shape of the input from `NHWC` to `NCHW`. Note that the onnx opset to be generated must be `12`.

次に、 `tflite` を `onnx` へ変換します。ここで `tensorflow-onnx` を使用します。 `--inputs-as-nchw input` は入力の形状を `NHWC` から `NCHW` へ変換するためのオプションです。なお、生成するonnxのopsetは `12` を指定する必要があります。

```
$ python -m tf2onnx.convert \
--opset 12 \
--inputs-as-nchw input \
--tflite ${MODEL}/saved_model_${H}x${W}/model_float32.tflite \
--output ${MODEL}/saved_model_${H}x${W}/model_float32.onnx
```

Redundant onnx files are generated with insufficient optimization and undefined input/output information for each operation.

最適化が不十分で、なおかつ各オペレーションの入出力情報が未定義の冗長なonnxファイルが生成されます。
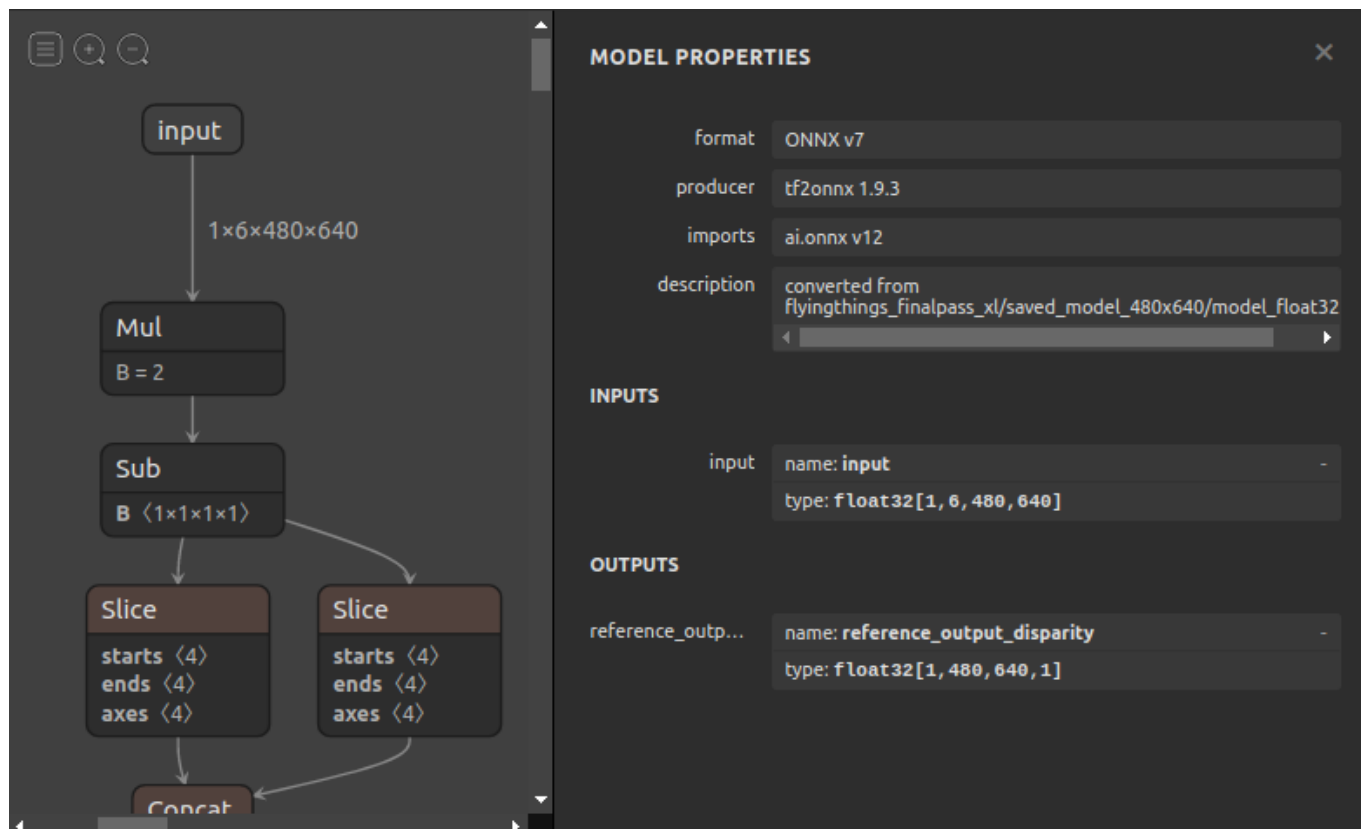
Uses `onnx-simplifier` to further optimize onnx files.

`onnx-simplifier` を使用してonnxファイルをさらに最適化します。

```
$ python -m onnxsim \
${MODEL}/saved_model_${H}x${W}/model_float32.onnx \
${MODEL}/saved_model_${H}x${W}/model_float32.onnx
```

The file size will increase, but the structure of the model will be optimized and inference performance will not be affected.

ファイルサイズが肥大化しますが、モデルの構造は最適化されおり推論パフォーマンスに影響はありません。
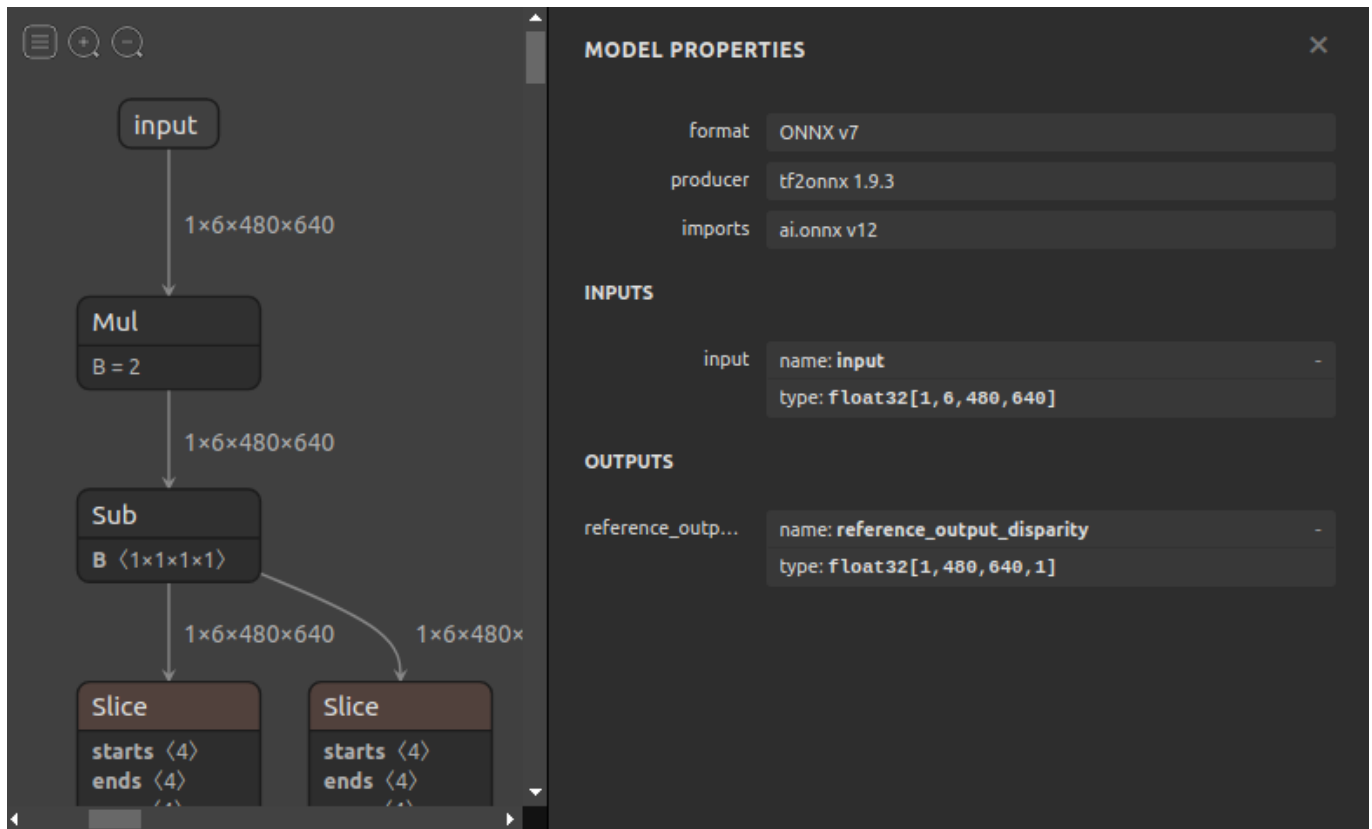
4-4. Building OpenVINO / OpenVINOのビルド

Since there are some issues with the current latest version of the OpenVINO model optimizer, we will build OpenVINO itself from the source code of the commits that have already resolved the issues. OpenVINOモデルオプティマイザの現行最新バージョンには一部問題があるため、問題箇所を解消済みのコミットのソースコードからOpenVINOそのものをビルドします。Intelのエンジニアとやりとりして解消いただいた問題点の内容が気になる方は こちら をご覧ください。

```
$ git clone https://github.com/openvinotoolkit/openvino \
&& cd openvino \
&& git checkout e89db1c6de8eb551949330114d476a2a4be499ed \
&& git submodule update --init --recursive \
&& pip install pip --upgrade \
&& pip install Cython numpy setuptools wheel pafy youtube-dl \
&& chmod +x scripts/submodule_update_with_gitee.sh \
&& ./scripts/submodule_update_with_gitee.sh \
&& chmod +x install_build_dependencies.sh \
&& ./install_build_dependencies.sh \
&& mkdir build \
&& cd build \
&& cmake \
-DCMAKE_BUILD_TYPE=Release \
-DENABLE_PYTHON=ON \
-DPYTHON_EXECUTABLE=`which python3` \
-DPYTHON_LIBRARY=/usr/lib/x86_64-linux-gnu/libpython3.8.so \
-DPYTHON_INCLUDE_DIR=/usr/include/python3.8 \
-DENABLE_CLDNN=ON \
```

```
    -DENABLE_WHEEL=ON .. \
    && make -j$(nproc)
```

Build finished.
ビルド終了。

```
adding 'openvino/test_utils/__init__.py'
adding 'openvino-2022.1.0.dist-info/LICENSE'
adding 'openvino-2022.1.0.dist-info/METADATA'
adding 'openvino-2022.1.0.dist-info/WHEEL'
adding 'openvino-2022.1.0.dist-info/onednn_third-party-programs.txt'
adding 'openvino-2022.1.0.dist-info/readme.txt'
adding 'openvino-2022.1.0.dist-info/runtime-third-party-programs.txt'
adding 'openvino-2022.1.0.dist-info/tbb_third-party-programs.txt'
adding 'openvino-2022.1.0.dist-info/top_level.txt'
adding 'openvino-2022.1.0.dist-info/RECORD'
removing build/bdist.linux-x86_64/wheel
[100%] Built target ie_wheel
user@ubuntu2004:~/workdir/openvino/build$
```

Check the generated Wheel files; two Wheel files have been generated.
生成されたWheelファイルを確認します。Wheelファイルは２個生成されています。

```
$ ls -l wheels/*
-rw-r--r-- 1 user user 30777895 Feb 11 11:17 wheels/openvino-2022.1.0-000-
cp38-cp38-manylinux_2_31_x86_64.whl
-rw-r--r-- 1 user user  6419721 Feb 11 11:06 wheels/openvino_dev-2022.1.0-
000-py3-none-any.whl
```

Overwrite the OpenVINO installation.
OpenVINOを上書きインストールします。

```
$ sudo ${INTEL_OPENVINO_DIR}/openvino_toolkit_uninstaller/uninstall.sh --
silent \
&& sudo pip install wheels/* && cd ../.. && rm -rf openvino
```

↑ Back to top

4-5. Convert ONNX to OpenVINO IR / ONNXをOpenVINO IRへ変換

Convert ONNX files to OpenVINO IR.
ONNXファイルをOpenVINO IRへ変換します。

```
$ sudo python /usr/local/lib/python3.8/dist-
packages/openvino/tools/mo/mo.py \
--input_model ${MODEL}/saved_model_${H}x${W}/model_float32.onnx \
--data_type FP32 \
--output_dir ${MODEL}/saved_model_${H}x${W}/openvino/FP32 \
```

```
--model_name ${MODEL}_${H}x${W} \
&& sudo chown -R user ${MODEL}
```

```
/usr/local/lib/python3.8/dist-packages/pkg_resources/__init__.py:122:
PkgResourcesDeprecationWarning: 0.1.9-nvc is an invalid version and will
not be supported in a future release
  warnings.warn(
/usr/local/lib/python3.8/dist-packages/pkg_resources/__init__.py:122:
PkgResourcesDeprecationWarning: 0.1.9-nvc is an invalid version and will
not be supported in a future release
  warnings.warn(
Model Optimizer arguments:
Common parameters:
    - Path to the Input Model:
/home/user/workdir/middlebury_d400/saved_model_480x640/model_float32.onnx
    - Path for generated IR:
/home/user/workdir/middlebury_d400/saved_model_480x640/openvino/FP32
    - IR output name:   middlebury_d400_480x640
    - Log level:    ERROR
    - Batch:    Not specified, inherited from the model
    - Input layers:    Not specified, inherited from the model
    - Output layers:    Not specified, inherited from the model
    - Input shapes:    Not specified, inherited from the model
    - Source layout:    Not specified
    - Target layout:    Not specified
    - Layout:   Not specified
    - Mean values:  Not specified
    - Scale values:    Not specified
    - Scale factor:    Not specified
    - Precision of IR:  FP32
    - Enable fusing:    True
    - Enable grouped convolutions fusing:   True
    - Move mean values to preprocess section:   None
    - Reverse input channels:   False
    - Use legacy API for model processing:  False
    - Use the transformations config file:  None
ONNX specific parameters:
/usr/local/lib/python3.8/dist-packages/pkg_resources/__init__.py:122:
PkgResourcesDeprecationWarning: 0.1.9-nvc is an invalid version and will
not be supported in a future release
  warnings.warn(
    - OpenVINO runtime found in:    /usr/local/lib/python3.8/dist-
packages/openvino
OpenVINO runtime version:
2022.1.custom_HEAD_e89db1c6de8eb551949330114d476a2a4be499ed
Model Optimizer version:
custom_HEAD_e89db1c6de8eb551949330114d476a2a4be499ed
[ WARNING ] Model Optimizer and OpenVINO runtime versions do no match.
[ WARNING ] Consider building the OpenVINO Python API from sources or
reinstall OpenVINO (TM) toolkit using "pip install openvino" (may be
incompatible with the current Model Optimizer version)
```

```
  [ WARNING ]
  Detected not satisfied dependencies:
      numpy: installed: 1.22.2, required: < 1.20
      fastjsonschema: not installed, required: ~= 2.15.1

  Please install required versions of components or use install_prerequisites
  script
  /usr/local/lib/python3.8/dist-
  packages/openvino/tools/mo/install_prerequisites/install_prerequisites_onnx
  .sh
  Note that install_prerequisites scripts may install additional components.
  /usr/local/lib/python3.8/dist-packages/pkg_resources/__init__.py:122:
  PkgResourcesDeprecationWarning: 0.1.9-nvc is an invalid version and will
  not be supported in a future release
    warnings.warn(
  [ SUCCESS ] Generated IR version 11 model.
  [ SUCCESS ] XML file:
  /home/user/workdir/middlebury_d400/saved_model_480x640/openvino/FP32/middle
  bury_d400_480x640.xml
  [ SUCCESS ] BIN file:
  /home/user/workdir/middlebury_d400/saved_model_480x640/openvino/FP32/middle
  bury_d400_480x640.bin
  [ SUCCESS ] Total execution time: 50.17 seconds.
  [ SUCCESS ] Memory consumed: 410 MB.
```
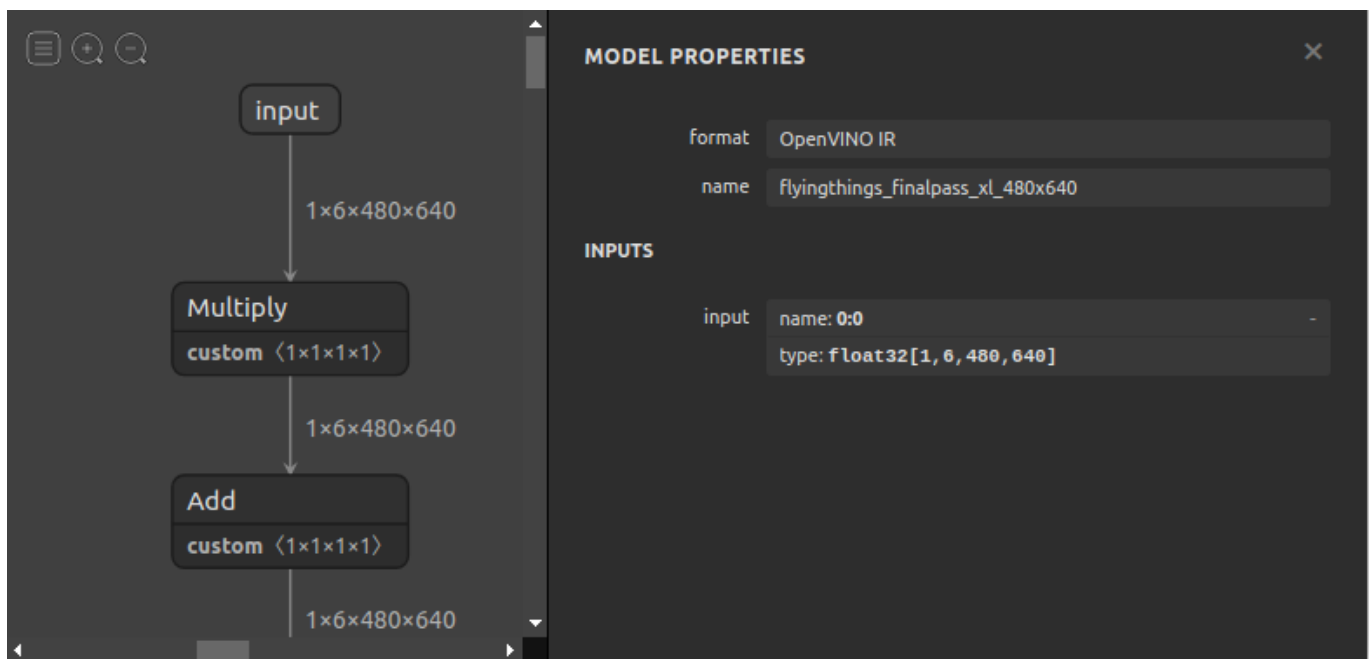

↑ Back to top

4-6. Download the Dataset / Datasetのダウンロード

Download a stereo driving dataset for testing. It is hard to see, but it is a dataset of pairs of images taken from each of the two left and right cameras.
テスト用のステレオドライビングデータセットをダウンロードします。見た目では分かりにくいですが、2個の左右のカメラからそれぞれ撮影した画像のペアのデータセットです。

Left	Right


```
$ mkdir -p "DrivingStereo images/left" \
&& mkdir -p "DrivingStereo images/right" \
&& mkdir -p "DrivingStereo images/depth" \
&& wget
https://github.com/PINTO0309/20210228_intel_deeplearning_day_hitnet_demo/re
leases/download/v1.0/2018-07-11-14-48-52_left.zip \
&& unzip -d "DrivingStereo images/left" -q 2018-07-11-14-48-52_left.zip \
&& rm 2018-07-11-14-48-52_left.zip \
&& wget
https://github.com/PINTO0309/20210228_intel_deeplearning_day_hitnet_demo/re
leases/download/v1.0/2018-07-11-14-48-52_right.zip \
&& unzip -d "DrivingStereo images/right" -q 2018-07-11-14-48-52_right.zip \
&& rm 2018-07-11-14-48-52_right.zip \
&& wget
https://github.com/PINTO0309/20210228_intel_deeplearning_day_hitnet_demo/re
leases/download/v1.0/2018-07-11-14-48-52_depth.zip \
&& unzip -d "DrivingStereo images/depth" -q 2018-07-11-14-48-52_depth.zip \
&& rm 2018-07-11-14-48-52_depth.zip \
&& wget
https://github.com/PINTO0309/20210228_intel_deeplearning_day_hitnet_demo/re
leases/download/v1.0/stereo_movie.mp4
```

4-7. HITNet's ONNX demo / HITNetのONNXデモ

4-7-1. ONNX+CUDA

I'll borrow ibaiGorordo's ONNX demo to run it. Adjust the program slightly so that ONNX's CUDA provider is enabled.
ibaiGorordoさんのONNXデモをお借りして実行してみます。ONNXのCUDAプロバイダが有効になるように、プログラムを少しだけ調整します。

```
$ rm -rf ONNX-HITNET-Stereo-Depth-estimation \
&& git clone https://github.com/ibaiGorordo/ONNX-HITNET-Stereo-Depth-
estimation.git \
&& cd ONNX-HITNET-Stereo-Depth-estimation \
&& git checkout 20471bfe2a23c34681141a9c0401eeff45680330 \
&& cd .. \
&& sed -i 's/models\///g' ONNX-HITNET-Stereo-Depth-
```

```
estimation/drivingStereoTest.py \
&& sed -i 's/cv2.WINDOW_NORMAL/cv2.WINDOW_AUTOSIZE/g' ONNX-HITNET-Stereo-
Depth-estimation/drivingStereoTest.py \
&& sed -i 's/max_distance = 30/max_distance = 80/g' ONNX-HITNET-Stereo-
Depth-estimation/drivingStereoTest.py \
&& sed -i 's/np.hstack((left_img,color_real_depth,
color_depth))/np.hstack((left_img, color_depth))/g' ONNX-HITNET-Stereo-
Depth-estimation/drivingStereoTest.py \
&& sed -i '31i \\t\tsession_option = onnxruntime.SessionOptions()' ONNX-
HITNET-Stereo-Depth-estimation/hitnet/hitnet.py \
&& sed -i '32i \\t\tmodel_file_name = model_path.split(".")[0]' ONNX-
HITNET-Stereo-Depth-estimation/hitnet/hitnet.py \
&& sed -i '33i \\t\tsession_option.optimized_model_filepath = f"
{model_file_name}_cudaopt.onnx"'  ONNX-HITNET-Stereo-Depth-
estimation/hitnet/hitnet.py \
&& sed -i '34i \\t\tsession_option.graph_optimization_level =
onnxruntime.GraphOptimizationLevel.ORT_ENABLE_EXTENDED' ONNX-HITNET-Stereo-
Depth-estimation/hitnet/hitnet.py \
&& sed -i
"s/onnxruntime.InferenceSession(model_path/onnxruntime.InferenceSession(mod
el_path, session_option, providers=[\'CUDAExecutionProvider\']/g" ONNX-
HITNET-Stereo-Depth-estimation/hitnet/hitnet.py

$ python ONNX-HITNET-Stereo-Depth-estimation/drivingStereoTest.py
```


4-7-2. ONNX+TensorRT

```
$ rm -rf ONNX-HITNET-Stereo-Depth-estimation \
&& git clone https://github.com/ibaiGorordo/ONNX-HITNET-Stereo-Depth-
estimation.git \
&& cd ONNX-HITNET-Stereo-Depth-estimation \
&& git checkout 20471bfe2a23c34681141a9c0401eeff45680330 \
&& cd .. \
&& sed -i 's/models\///g' ONNX-HITNET-Stereo-Depth-
estimation/drivingStereoTest.py \
&& sed -i 's/cv2.WINDOW_NORMAL/cv2.WINDOW_AUTOSIZE/g' ONNX-HITNET-Stereo-
Depth-estimation/drivingStereoTest.py \
&& sed -i 's/max_distance = 30/max_distance = 80/g' ONNX-HITNET-Stereo-
Depth-estimation/drivingStereoTest.py \
&& sed -i 's/np.hstack((left_img,color_real_depth,
```

```
color_depth))/np.hstack((left_img, color_depth))/g' ONNX-HITNET-Stereo-
Depth-estimation/drivingStereoTest.py \
&& sed -i
"s/onnxruntime.InferenceSession(model_path/onnxruntime.InferenceSession(mod
el_path, providers=[\'TensorrtExecutionProvider',
'CUDAExecutionProvider']/g" ONNX-HITNET-Stereo-Depth-
estimation/hitnet/hitnet.py
```

https://user-images.githubusercontent.com/33194443/156153624-4a94754e-bfaf-470f-a830-2c9483efb474.mp4

↑ Back to top

4-8. HITNet's OpenVINO demo / HITNetのOpenVINOデモ

Run a test inference program customized for OpenVINO: CPU inference.
OpenVINO用にカスタマイズしたテスト用推論プログラムを実行します。CPU推論です。

```
$ python drivingStereoTest_openvino.py
```


↑ Back to top

4-9. HITNet's TensorRT demo / HITNetのTensorRTデモ

I will be borrowing iwatake's TensorRT demo to run the test. Follow the tutorial in this repository to set up and run the environment.
iwatakeさんのTensorRTデモをお借りしてテストを実施します。こちらのリポジトリのチュートリアルに従って環境を構築して実行します。
https://github.com/iwatake2222/play_with_tensorrt/tree/master/pj_tensorrt_depth_stereo_hitnet

```
$ ./main stereo_movie.mp4
```

↑ Back to top

5. Acknowledgements / 謝辞

Thanks!!!

- **Intel Team**:

  - GatherND shape conversion from ONNX is inaccurate #7379
  - Const data got different desc and content byte sizes (24 and 96 respectively)" error when converting ConvolutionBackpropData using compile_tools #9517

- **openvinotoolkit**:

  - https://github.com/openvinotoolkit/openvino

  - ▶ LICENSE

    ```
            Apache License
          Version 2.0, January 2004
       http://www.apache.org/licenses/

     TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
    ```

     1. Definitions.

        "License" shall mean the terms and conditions for use,
reproduction,
        and distribution as defined by Sections 1 through 9 of this
document.

        "Licensor" shall mean the copyright owner or entity
authorized by
        the copyright owner that is granting the License.

        "Legal Entity" shall mean the union of the acting entity
and all
        other entities that control, are controlled by, or are
under common
        control with that entity. For the purposes of this
definition,
        "control" means (i) the power, direct or indirect, to cause
the
        direction or management of such entity, whether by contract
or
        otherwise, or (ii) ownership of fifty percent (50%) or more
of the
        outstanding shares, or (iii) beneficial ownership of such
entity.

        "You" (or "Your") shall mean an individual or Legal Entity
        exercising permissions granted by this License.

        "Source" form shall mean the preferred form for making
modifications,
        including but not limited to software source code,
documentation
        source, and configuration files.

        "Object" form shall mean any form resulting from mechanical
        transformation or translation of a Source form, including
but
        not limited to compiled object code, generated
documentation,
        and conversions to other media types.

        "Work" shall mean the work of authorship, whether in Source
or
        Object form, made available under the License, as indicated
by a
        copyright notice that is included in or attached to the
work
        (an example is provided in the Appendix below).

        "Derivative Works" shall mean any work, whether in Source
or Object
        form, that is based on (or derived from) the Work and for

which the
        editorial revisions, annotations, elaborations, or other
modifications
        represent, as a whole, an original work of authorship. For
the purposes
        of this License, Derivative Works shall not include works
that remain
        separable from, or merely link (or bind by name) to the
interfaces of,
        the Work and Derivative Works thereof.

        "Contribution" shall mean any work of authorship, including
        the original version of the Work and any modifications or
additions
        to that Work or Derivative Works thereof, that is
intentionally
        submitted to Licensor for inclusion in the Work by the
copyright owner
        or by an individual or Legal Entity authorized to submit on
behalf of
        the copyright owner. For the purposes of this definition,
"submitted"
        means any form of electronic, verbal, or written
communication sent
        to the Licensor or its representatives, including but not
limited to
        communication on electronic mailing lists, source code
control systems,
        and issue tracking systems that are managed by, or on
behalf of, the
        Licensor for the purpose of discussing and improving the
Work, but
        excluding communication that is conspicuously marked or
otherwise
        designated in writing by the copyright owner as "Not a
Contribution."

        "Contributor" shall mean Licensor and any individual or
Legal Entity
        on behalf of whom a Contribution has been received by
Licensor and
        subsequently incorporated within the Work.

   2. Grant of Copyright License. Subject to the terms and
conditions of
        this License, each Contributor hereby grants to You a
perpetual,
        worldwide, non-exclusive, no-charge, royalty-free,
irrevocable
        copyright license to reproduce, prepare Derivative Works
of,
        publicly display, publicly perform, sublicense, and
distribute the
        Work and such Derivative Works in Source or Object form.

    3. Grant of Patent License. Subject to the terms and
conditions of
      this License, each Contributor hereby grants to You a
perpetual,
      worldwide, non-exclusive, no-charge, royalty-free,
irrevocable
      (except as stated in this section) patent license to make,
have made,
      use, offer to sell, sell, import, and otherwise transfer
the Work,
      where such license applies only to those patent claims
licensable
      by such Contributor that are necessarily infringed by their
      Contribution(s) alone or by combination of their
Contribution(s)
      with the Work to which such Contribution(s) was submitted.
If You
      institute patent litigation against any entity (including a
      cross-claim or counterclaim in a lawsuit) alleging that the
Work
      or a Contribution incorporated within the Work constitutes
direct
      or contributory patent infringement, then any patent
licenses
      granted to You under this License for that Work shall
terminate
      as of the date such litigation is filed.

    4. Redistribution. You may reproduce and distribute copies of
the
      Work or Derivative Works thereof in any medium, with or
without
      modifications, and in Source or Object form, provided that
You
      meet the following conditions:

      (a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

      (b) You must cause any modified files to carry prominent
notices
    stating that You changed the files; and

      (c) You must retain, in the Source form of any Derivative
Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

      (d) If the Work includes a "NOTICE" text file as part of
its
    distribution, then any Derivative Works that You distribute

must
    include a readable copy of the attribution notices contained
    within such NOTICE file, excluding those notices that do not
    pertain to any part of the Derivative Works, in at least one
    of the following places: within a NOTICE text file
distributed
    as part of the Derivative Works; within the Source form or
    documentation, if provided along with the Derivative Works;
or,
    within a display generated by the Derivative Works, if and
    wherever such third-party notices normally appear. The
contents
    of the NOTICE file are for informational purposes only and
    do not modify the License. You may add Your own attribution
    notices within Derivative Works that You distribute,
alongside
    or as an addendum to the NOTICE text from the Work, provided
    that such additional attribution notices cannot be construed
    as modifying the License.

    You may add Your own copyright statement to Your
modifications and
    may provide additional or different license terms and
conditions
    for use, reproduction, or distribution of Your
modifications, or
    for any such Derivative Works as a whole, provided Your
use,
    reproduction, and distribution of the Work otherwise
complies with
    the conditions stated in this License.

  5. Submission of Contributions. Unless You explicitly state
otherwise,
    any Contribution intentionally submitted for inclusion in
the Work
    by You to the Licensor shall be under the terms and
conditions of
    this License, without any additional terms or conditions.
    Notwithstanding the above, nothing herein shall supersede
or modify
    the terms of any separate license agreement you may have
executed
    with Licensor regarding such Contributions.

  6. Trademarks. This License does not grant permission to use
the trade
    names, trademarks, service marks, or product names of the
Licensor,
    except as required for reasonable and customary use in
describing the
    origin of the Work and reproducing the content of the
NOTICE file.

      7. Disclaimer of Warranty. Unless required by applicable law
or
      agreed to in writing, Licensor provides the Work (and each
      Contributor provides its Contributions) on an "AS IS"
BASIS,
      WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or
      implied, including, without limitation, any warranties or
conditions
      of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR
A
      PARTICULAR PURPOSE. You are solely responsible for
determining the
      appropriateness of using or redistributing the Work and
assume any
      risks associated with Your exercise of permissions under
this License.

   8. Limitation of Liability. In no event and under no legal
theory,
      whether in tort (including negligence), contract, or
otherwise,
      unless required by applicable law (such as deliberate and
grossly
      negligent acts) or agreed to in writing, shall any
Contributor be
      liable to You for damages, including any direct, indirect,
special,
      incidental, or consequential damages of any character
arising as a
      result of this License or out of the use or inability to
use the
      Work (including but not limited to damages for loss of
goodwill,
      work stoppage, computer failure or malfunction, or any and
all
      other commercial damages or losses), even if such
Contributor
      has been advised of the possibility of such damages.

   9. Accepting Warranty or Additional Liability. While
redistributing
      the Work or Derivative Works thereof, You may choose to
offer,
      and charge a fee for, acceptance of support, warranty,
indemnity,
      or other liability obligations and/or rights consistent
with this
      License. However, in accepting such obligations, You may
act only
      on Your own behalf and on Your sole responsibility, not on
behalf
      of any other Contributor, and only if You agree to
indemnify,

- **NobuoTsukamoto**:

  - https://github.com/NobuoTsukamoto/benchmarks

  ▶ LICENSE

- **ibaiGorordo**:

  - https://github.com/ibaiGorordo/ONNX-HITNET-Stereo-Depth-estimation

    ▶ LICENSE

- **iwatake2222**:

    - https://github.com/iwatake2222/play_with_tensorrt

    ▶ LICENSE

```
    "control" means (i) the power, direct or indirect, to cause
the
    direction or management of such entity, whether by contract
or
    otherwise, or (ii) ownership of fifty percent (50%) or more
of the
    outstanding shares, or (iii) beneficial ownership of such
entity.

    "You" (or "Your") shall mean an individual or Legal Entity
    exercising permissions granted by this License.

    "Source" form shall mean the preferred form for making
modifications,
    including but not limited to software source code,
documentation
    source, and configuration files.

    "Object" form shall mean any form resulting from mechanical
    transformation or translation of a Source form, including but
    not limited to compiled object code, generated documentation,
    and conversions to other media types.

    "Work" shall mean the work of authorship, whether in Source
or
    Object form, made available under the License, as indicated
by a
    copyright notice that is included in or attached to the work
    (an example is provided in the Appendix below).

    "Derivative Works" shall mean any work, whether in Source or
Object
    form, that is based on (or derived from) the Work and for
which the
    editorial revisions, annotations, elaborations, or other
modifications
    represent, as a whole, an original work of authorship. For
the purposes
    of this License, Derivative Works shall not include works
that remain
    separable from, or merely link (or bind by name) to the
interfaces of,
    the Work and Derivative Works thereof.

    "Contribution" shall mean any work of authorship, including
    the original version of the Work and any modifications or
additions
    to that Work or Derivative Works thereof, that is
intentionally
    submitted to Licensor for inclusion in the Work by the
copyright owner
    or by an individual or Legal Entity authorized to submit on
behalf of
    the copyright owner. For the purposes of this definition,
```

"submitted"
    means any form of electronic, verbal, or written
communication sent
    to the Licensor or its representatives, including but not
limited to
    communication on electronic mailing lists, source code
control systems,
    and issue tracking systems that are managed by, or on behalf
of, the
    Licensor for the purpose of discussing and improving the
Work, but
    excluding communication that is conspicuously marked or
otherwise
    designated in writing by the copyright owner as "Not a
Contribution."

    "Contributor" shall mean Licensor and any individual or Legal
Entity
    on behalf of whom a Contribution has been received by
Licensor and
    subsequently incorporated within the Work.

 2. Grant of Copyright License. Subject to the terms and
conditions of
    this License, each Contributor hereby grants to You a
perpetual,
    worldwide, non-exclusive, no-charge, royalty-free,
irrevocable
    copyright license to reproduce, prepare Derivative Works of,
    publicly display, publicly perform, sublicense, and
distribute the
    Work and such Derivative Works in Source or Object form.

 3. Grant of Patent License. Subject to the terms and conditions
of
    this License, each Contributor hereby grants to You a
perpetual,
    worldwide, non-exclusive, no-charge, royalty-free,
irrevocable
    (except as stated in this section) patent license to make,
have made,
    use, offer to sell, sell, import, and otherwise transfer the
Work,
    where such license applies only to those patent claims
licensable
    by such Contributor that are necessarily infringed by their
    Contribution(s) alone or by combination of their
Contribution(s)
    with the Work to which such Contribution(s) was submitted. If
You
    institute patent litigation against any entity (including a
    cross-claim or counterclaim in a lawsuit) alleging that the
Work
    or a Contribution incorporated within the Work constitutes

direct
    or contributory patent infringement, then any patent licenses
    granted to You under this License for that Work shall
terminate
    as of the date such litigation is filed.

 4. Redistribution. You may reproduce and distribute copies of
the
    Work or Derivative Works thereof in any medium, with or
without
    modifications, and in Source or Object form, provided that
You
    meet the following conditions:

    (a) You must give any other recipients of the Work or
        Derivative Works a copy of this License; and

    (b) You must cause any modified files to carry prominent
notices
        stating that You changed the files; and

    (c) You must retain, in the Source form of any Derivative
Works
        that You distribute, all copyright, patent, trademark,
and
        attribution notices from the Source form of the Work,
        excluding those notices that do not pertain to any part
of
        the Derivative Works; and

    (d) If the Work includes a "NOTICE" text file as part of its
        distribution, then any Derivative Works that You
distribute must
        include a readable copy of the attribution notices
contained
        within such NOTICE file, excluding those notices that do
not
        pertain to any part of the Derivative Works, in at least
one
        of the following places: within a NOTICE text file
distributed
        as part of the Derivative Works; within the Source form
or
        documentation, if provided along with the Derivative
Works; or,
        within a display generated by the Derivative Works, if
and
        wherever such third-party notices normally appear. The
contents
        of the NOTICE file are for informational purposes only
and
        do not modify the License. You may add Your own
attribution
        notices within Derivative Works that You distribute,

alongside
        or as an addendum to the NOTICE text from the Work,
provided
        that such additional attribution notices cannot be
construed
        as modifying the License.

    You may add Your own copyright statement to Your
modifications and
    may provide additional or different license terms and
conditions
    for use, reproduction, or distribution of Your modifications,
or
    for any such Derivative Works as a whole, provided Your use,
    reproduction, and distribution of the Work otherwise complies
with
    the conditions stated in this License.

 5. Submission of Contributions. Unless You explicitly state
otherwise,
    any Contribution intentionally submitted for inclusion in the
Work
    by You to the Licensor shall be under the terms and
conditions of
    this License, without any additional terms or conditions.
    Notwithstanding the above, nothing herein shall supersede or
modify
    the terms of any separate license agreement you may have
executed
    with Licensor regarding such Contributions.

 6. Trademarks. This License does not grant permission to use the
trade
    names, trademarks, service marks, or product names of the
Licensor,
    except as required for reasonable and customary use in
describing the
    origin of the Work and reproducing the content of the NOTICE
file.

 7. Disclaimer of Warranty. Unless required by applicable law or
    agreed to in writing, Licensor provides the Work (and each
    Contributor provides its Contributions) on an "AS IS" BASIS,
    WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
or
    implied, including, without limitation, any warranties or
conditions
    of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
    PARTICULAR PURPOSE. You are solely responsible for
determining the
    appropriateness of using or redistributing the Work and
assume any
    risks associated with Your exercise of permissions under this
License.

- **A Large-Scale Dataset for Stereo Matching in Autonomous Driving Scenarios**:

  - https://drivingstereo-dataset.github.io/

    ▶ LICENSE

```
@inproceedings{yang2019drivingstereo,
    title={DrivingStereo: A Large-Scale Dataset for Stereo
Matching in Autonomous Driving Scenarios},
    author={Yang, Guorun and Song, Xiao and Huang, Chaoqin and
Deng, Zhidong and Shi, Jianping and Zhou, Bolei},
    booktitle={IEEE Conference on Computer Vision and Pattern
Recognition (CVPR)},
    year={2019}
}
```