

MATLAB® & SIMULINK®

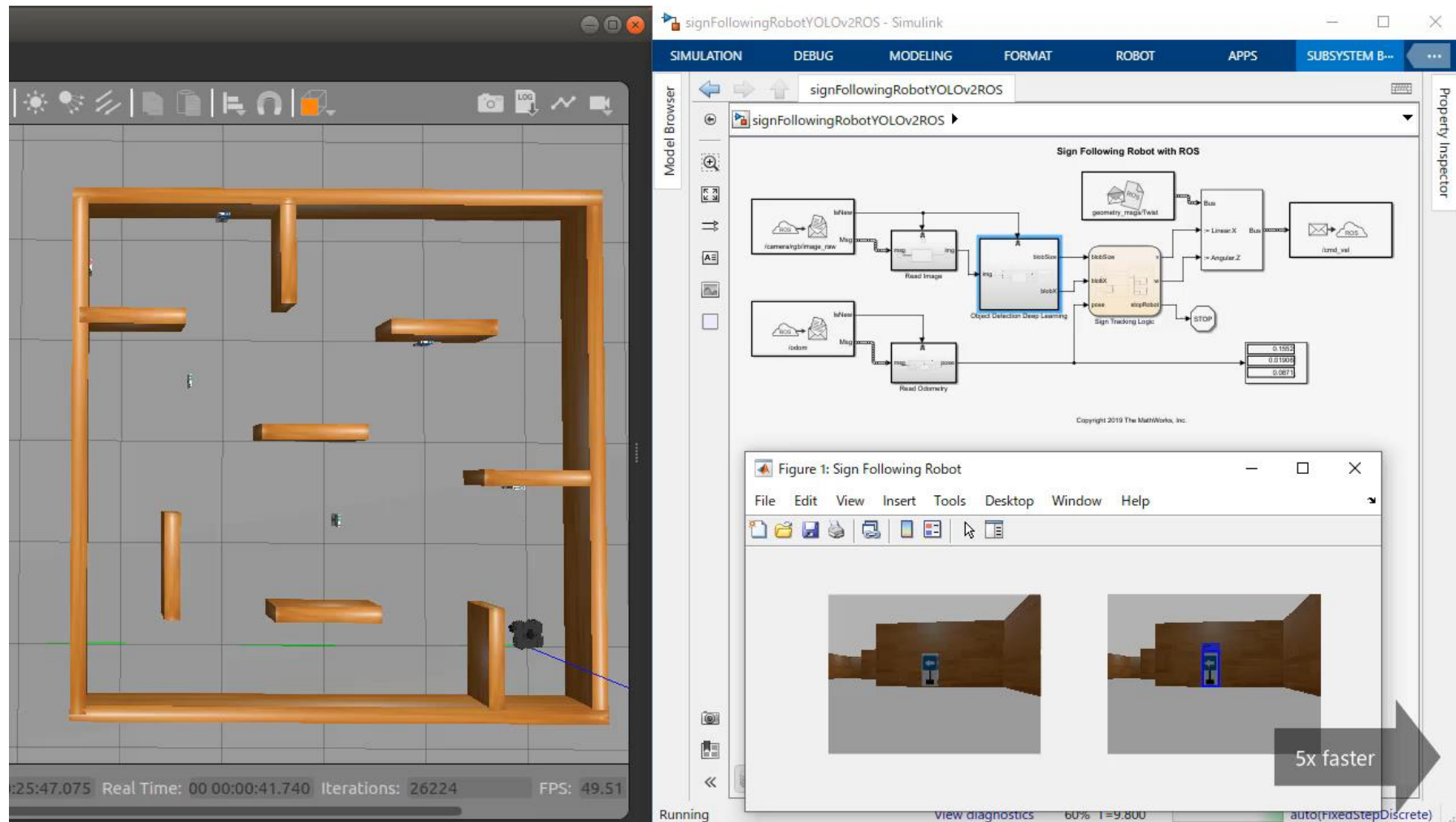


無料体験！MATLAB および Simulink によるAI・自律ロボティクス開発入門

Deep Learning による物体認識



デモ：ディープラーニング物体検出器によるロボット制御



ロボット制御に活用可能なディープラーニングの画像認識モデルを設計

ワークショップの準備

1. "MATLAB R2021a" を起動



2. デモファイルのあるフォルダに移動

```
>> cd 1_perception_deep_learning
```

3. ファイルを開く

```
>> trainYOLOv2ObjectDetector_jp.mlx
```

ディープラーニング領域での MATLAB の進化

2016

畳み込みニューラル
ネットワーク
学習済みモデル
Caffeインポーター

2017

名称変更

- Neural Network Toolbox
からDeep Learning
Toolboxへ

アルゴリズム

- LSTM
- Directed Acyclic Graphs
(DAGネットワーク)
- マルチGPU 学習

コード生成

- GPU Coder

アプリ

- イメージラベラー

サードパーティ連携

- TensorFlow-Keras
インポーター

2018

リファレンスサンプル

- 信号処理
- オーディオ
- テキスト解析

アルゴリズム

- ウェブレットスキッタ
リング
- 大規模画像(Big Image)

コード生成

- MATLAB Coder C++

アプリ

- ディープネットワーク
デザイナー
- ビデオラベラー
- オーディオラベラー

サードパーティ連携

- ONNX サポート

2019

強化学習サポート

アルゴリズム

- 自動微分
- カスタム学習ループ
- 重み共有

リファレンスサンプル

- YOLOv2
- GAN
- Siamese Network
- オートエンコーダー(VAE)
- 3-D サポート

説明可能な AI

- Occlusion
- Grad-CAM

コード生成

- MATLAB Coder (ARM)

アプリ

- シグナルラベラー

2020

- ディープラーニング
データセット

アプリ

- 実験マネージャー
- Lidar Labeler

リファレンスサンプル

- 5G通信
- Lidar
- 200以上の例題

アルゴリズム

- 点群(PointNet)
- SSD

コード生成

- 量子化
- HDLコード生成・
FPGA実装

コード生成

- Simulinkでの
ディープラーニング

2021

- 250+以上の例題

アプリ

- 強化学習デザイナー
- モデルベースデザイン

- Simulinkによる
再帰ニューラル
ネットワーク

アルゴリズム

- 高次微分

例題

- 連合学習
- ニューラルODE
- グラフ畳み込み
ネットワーク
- トランスフォーマーモデル*

サードパーティ連携

- TensorFlow モデル
インポーター

MATLABがサポートするディープラーニングワークフロー

ディープラーニングを効率的に学習し、システムに統合


データ収集

 データの
クレンジングや
前処理

 ドメインナレッジ

 シミュレーションに
よるデータの水増し


AI モデリング

 モデルの設計と
チューニング

 ハードウェアによ
る学習の高速化

 他フレームワーク
連携

システムデザイン

 複合的なシステムへ
の統合

 システム
シミュレーション

 システムの検証

展開

 組み込みデバイス

 エンタープライズ
システム

 エッジ, クラウド,
デスクトップ

画像処理とディープラーニング

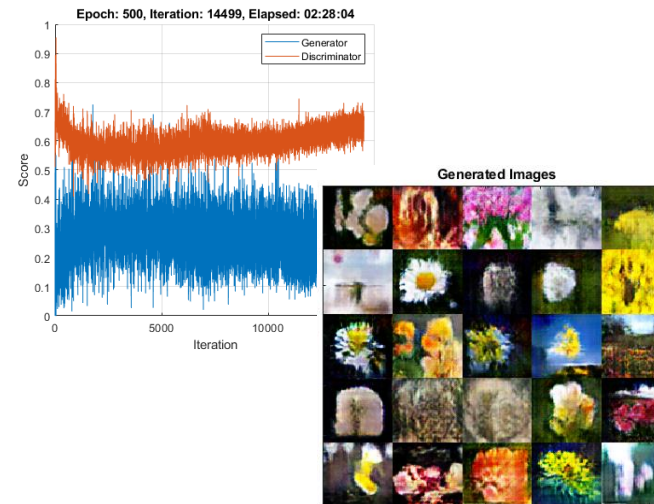
YOLO v3



Object Detection Using
YOLO v3 Deep Learning
*Computer Vision Toolbox™
Deep Learning Toolbox™*

R2020a

GAN

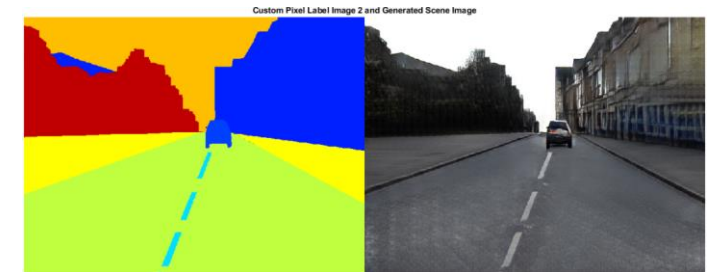


Train Generative Adversarial
Network (GAN)

*Deep Learning Toolbox™
Parallel Computing Toolbox™*

R2019b

Pix2PixHD



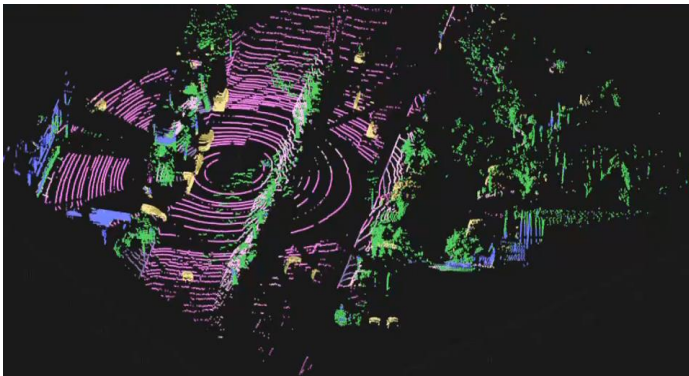
Generate Image from
Segmentation Map Using
Deep Learning

*Computer Vision Toolbox™
Deep Learning Toolbox™*

R2020b

点群処理とディープラーニング

セマンティック セグメンテーション



Lidar Point Cloud Semantic
Segmentation Using SqueezeSegV2
Deep Learning Network

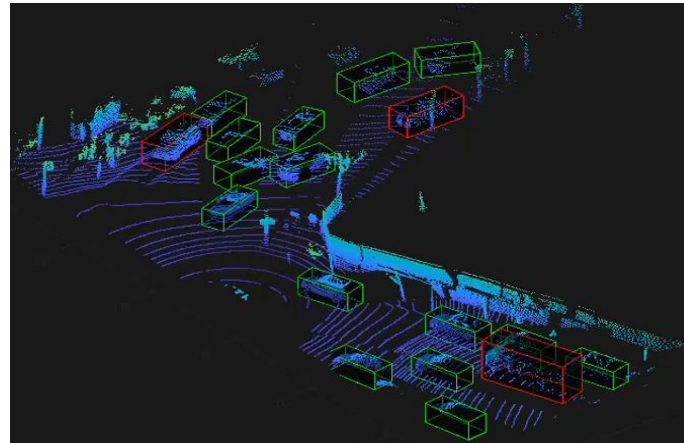
Computer Vision Toolbox™

Deep Learning Toolbox™

Lidar Toolbox™

R2021a

物体検出



Lidar 3-D Object Detection
Using PointPillars Deep
Learning

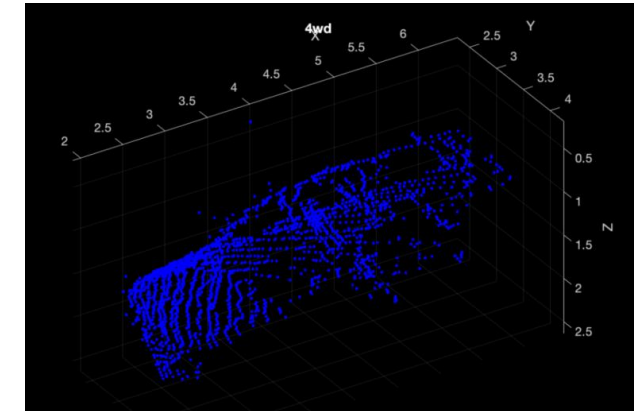
Computer Vision Toolbox™

Deep Learning Toolbox™

Lidar Toolbox™

R2021a

分類



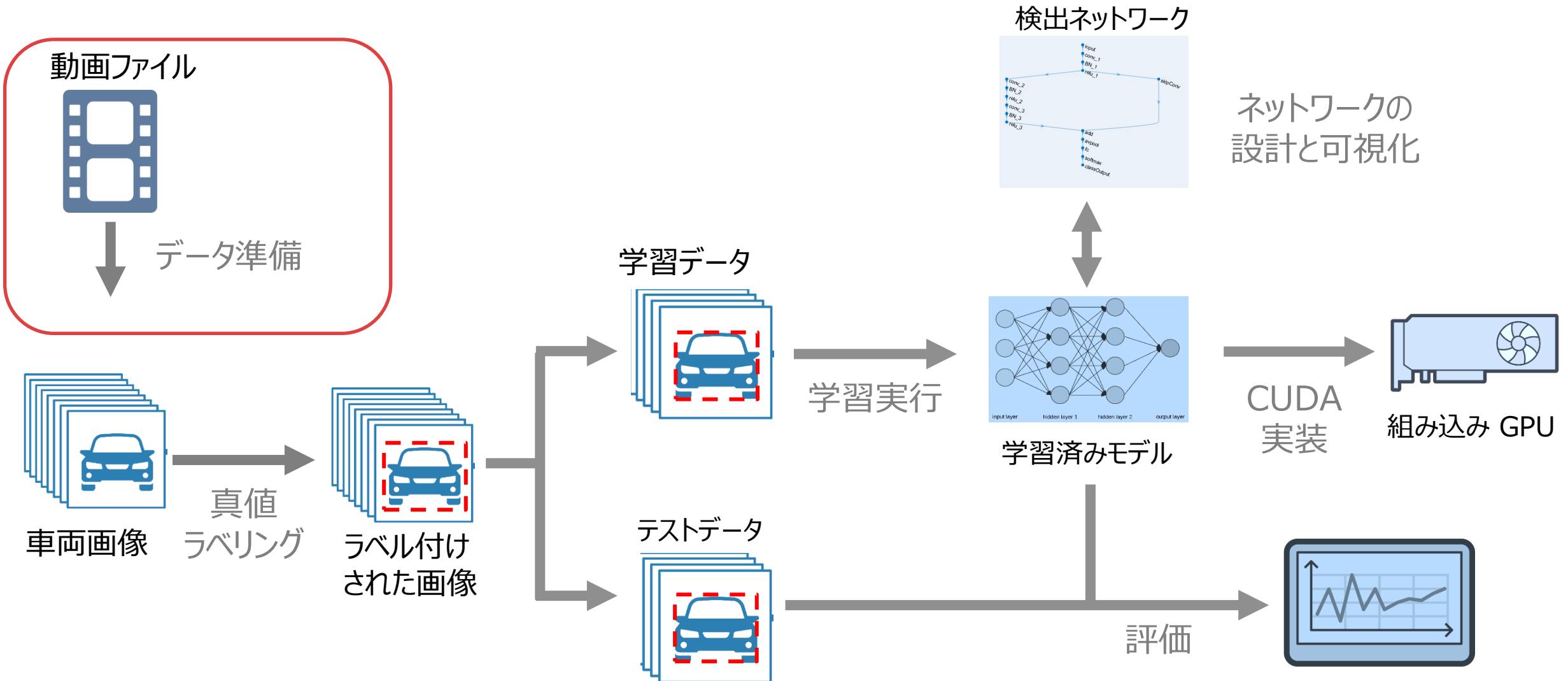
Point Cloud Classification
Using PointNet Deep Learning

Computer Vision Toolbox™

Deep Learning Toolbox™

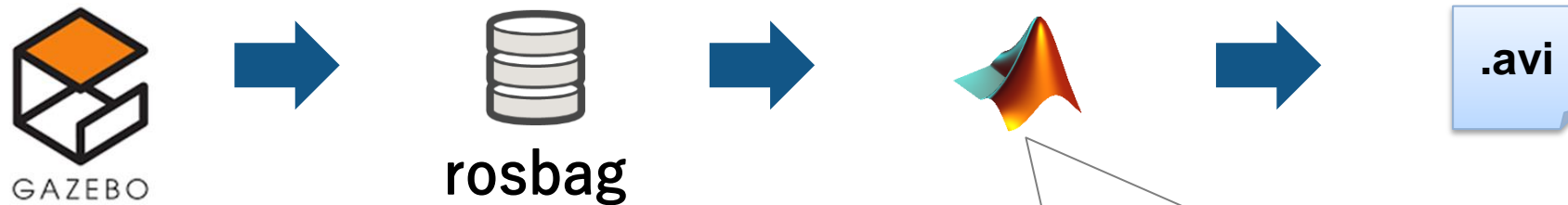
R2020a

YOLO v2 物体検出の学習



学習用画像の準備

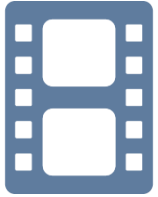
- 学習用画像を静止画もしくは動画として収集
- 例
 - Gazeboでカメラシミュレーションを行い、rosvagとして保存
 - rosvagをMATLABで開き、動画ファイルとしてエクスポート



```
bagselect = rosvag('turtlebot3.bag');  
bagselectImage = ...  
    select(bagselect, 'Topic', '/camera/rgb/image_raw');  
msgStructs = ...  
    readMessages(bagselectImage, 1, ...  
        "DataFormat", "struct"); % 1番目のメッセージ取り出し
```

YOLO v2: 物体検出のワークフロー

動画ファイル



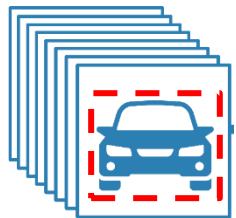
データ準備



車両画像

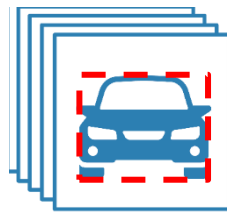
真値

ラベリング



ラベル付け
された画像

学習データ

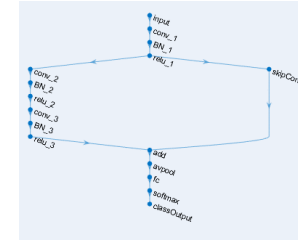


学習実行

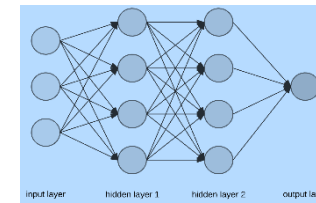
テストデータ



検出ネットワーク

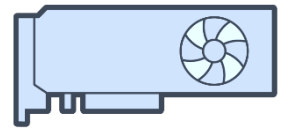


ネットワークの
設計と可視化



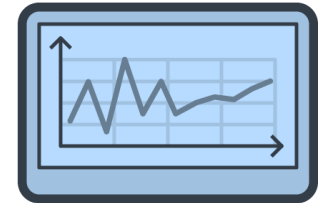
学習済みモデル

CUDA
実装



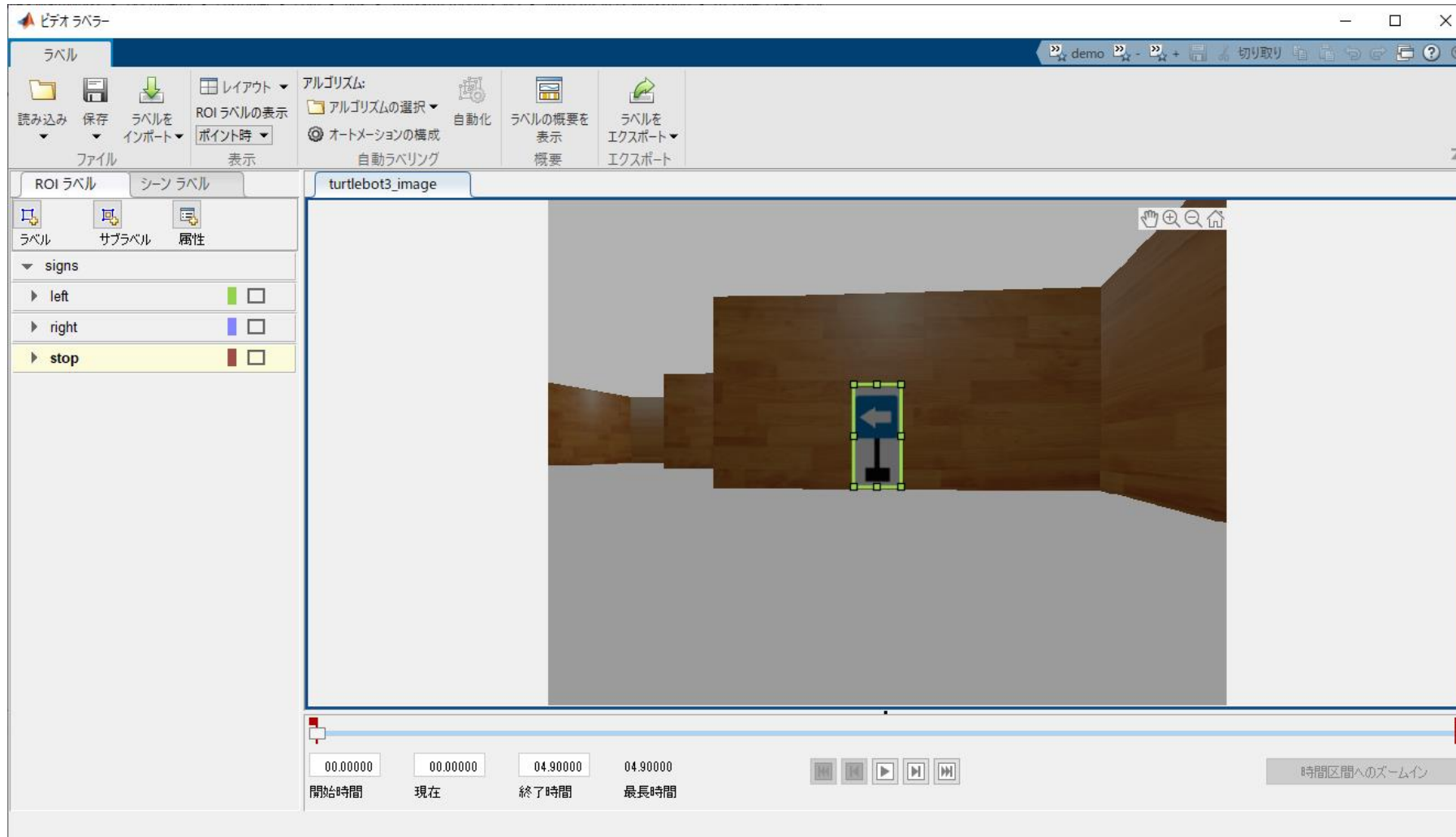
組み込み GPU

評価

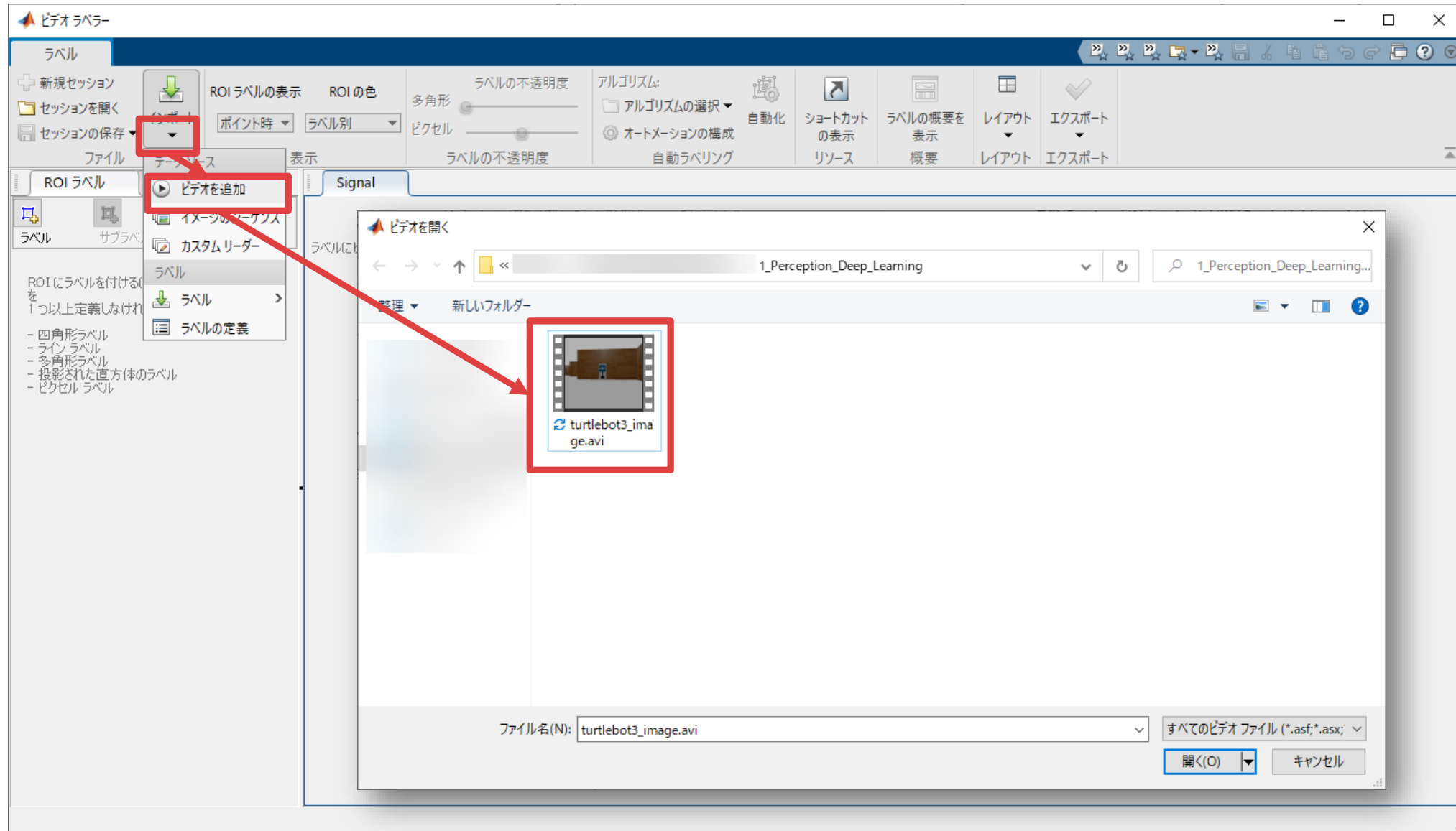


学習データのラベリング

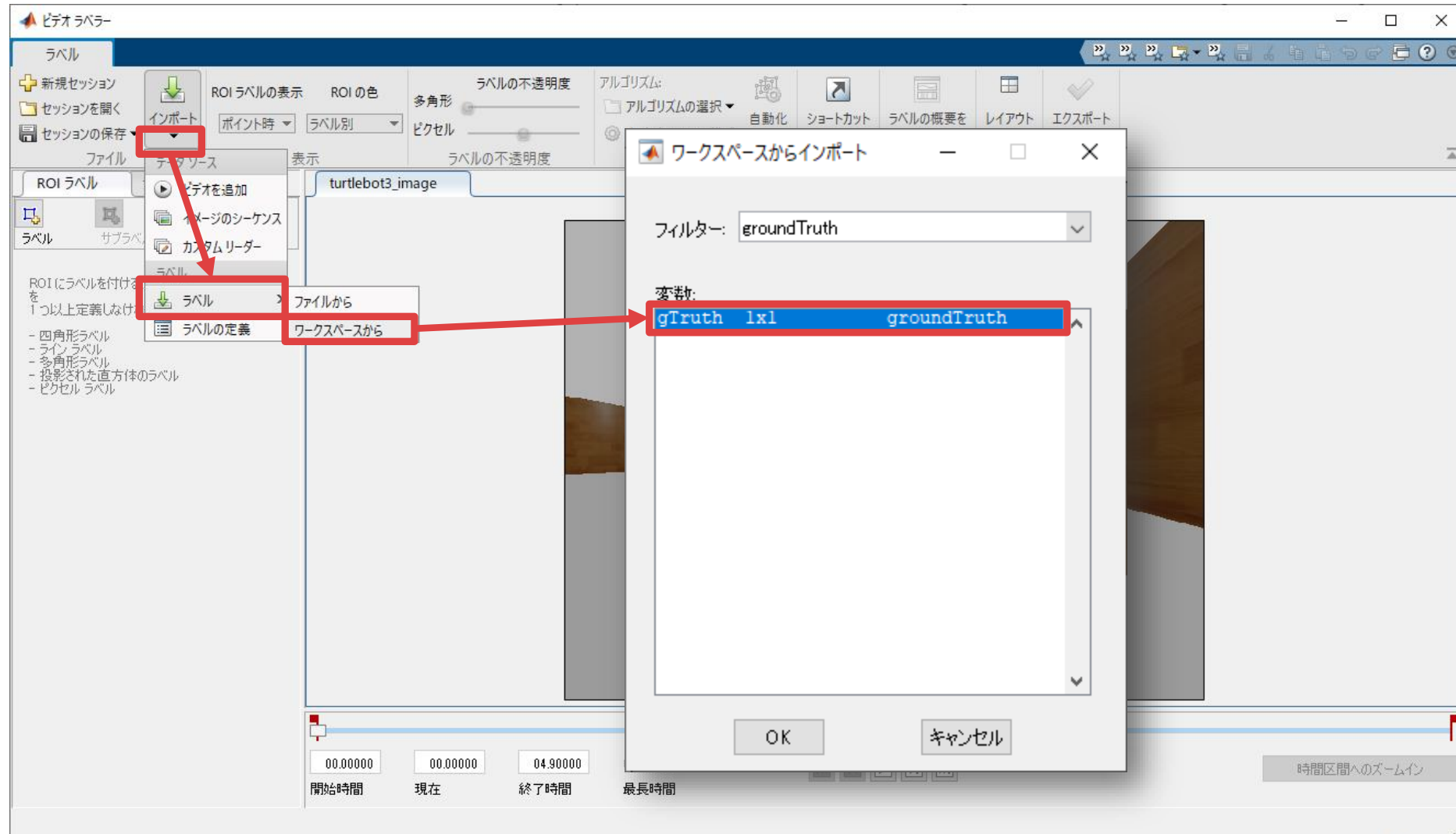
```
videoLabeler('turtlebot3_image.avi');
```



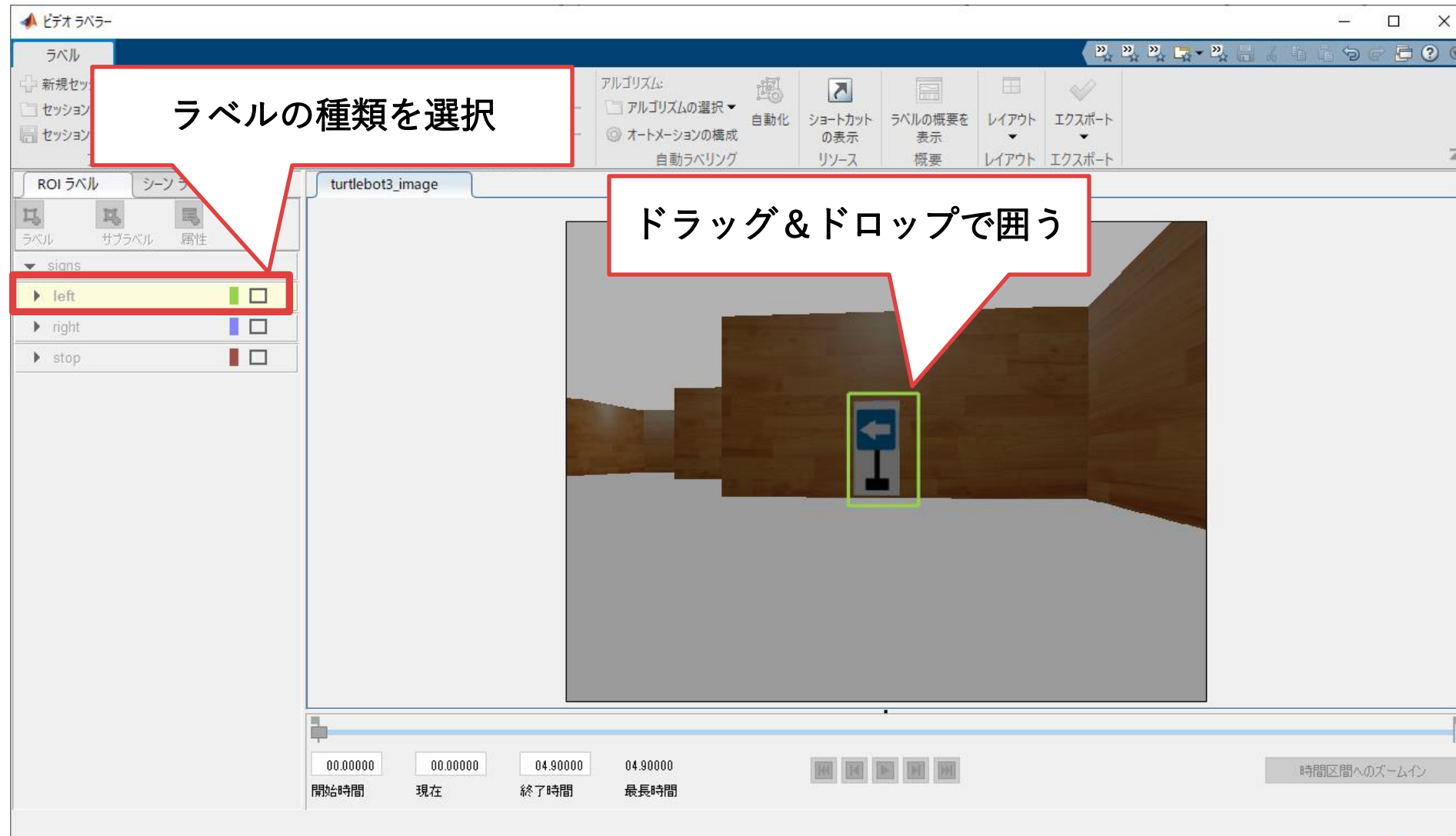
ビデオファイルを開く



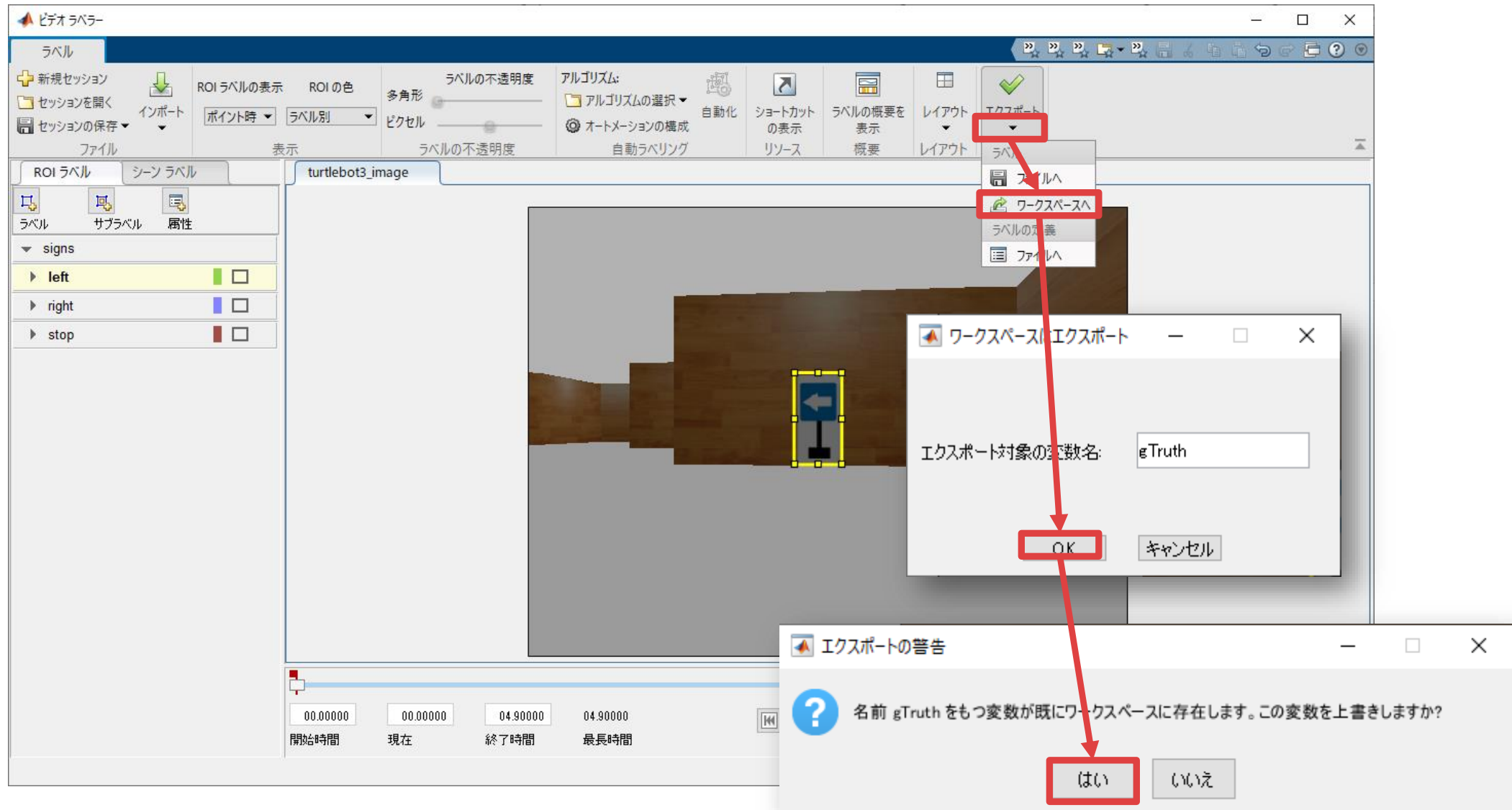
事前にラベル付けしておいたデータを開く



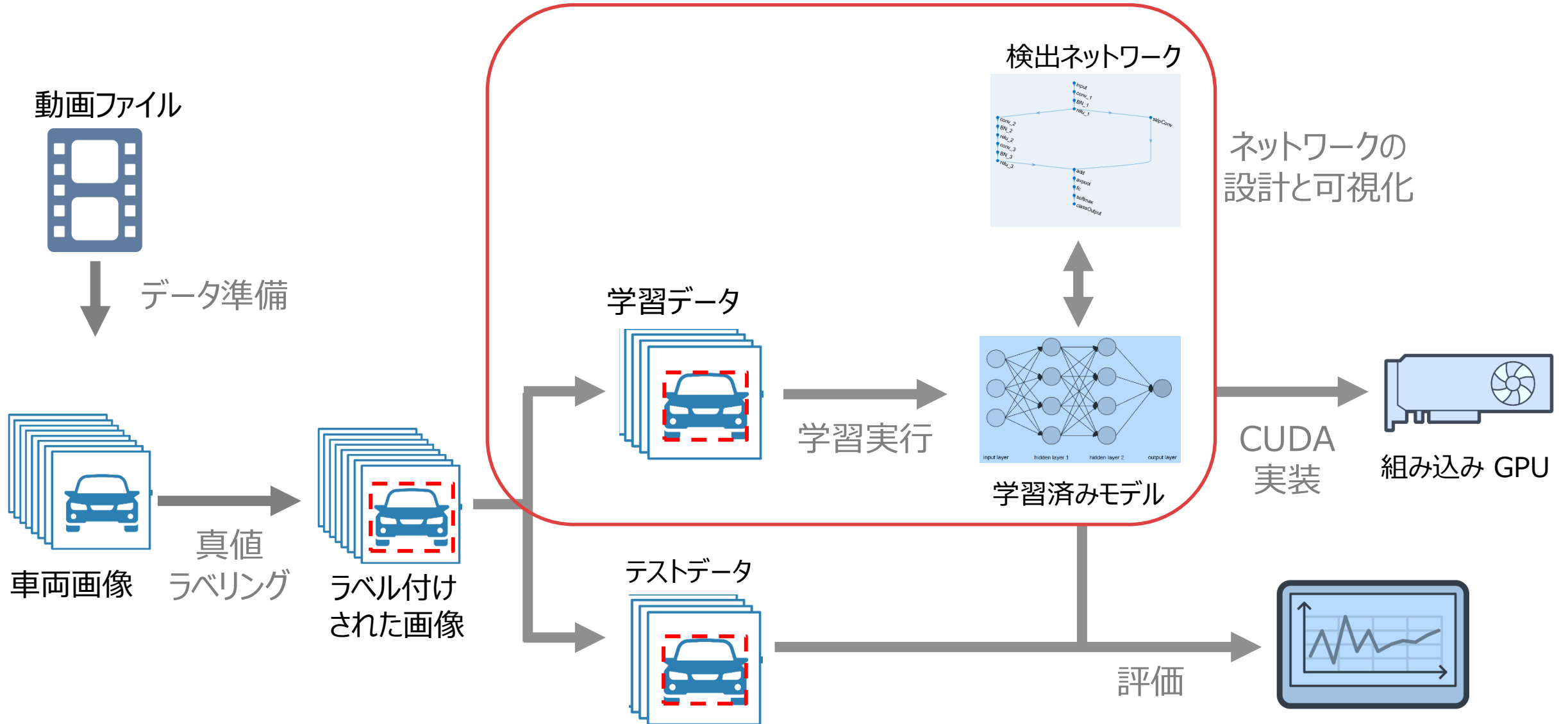
ラベル付けを行う



結果をエクスポート

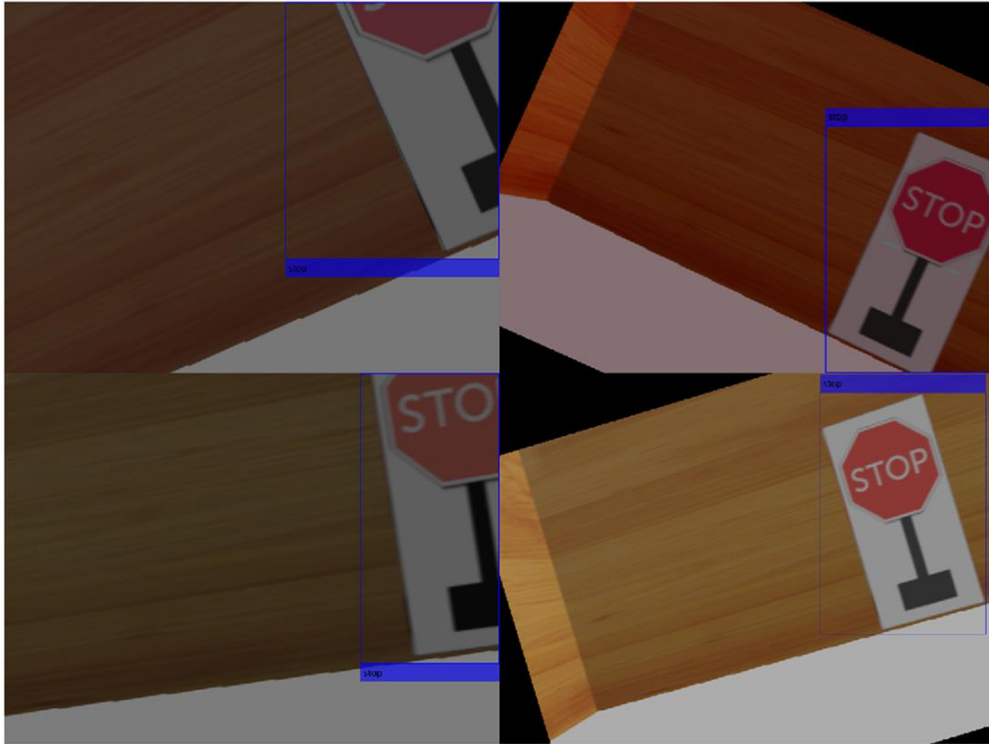


物体検出のワークフロー



学習データの増強と正規化

Deep Learning Toolbox
Computer Vision Toolbox



色味、回転、拡大縮小を人工的に加えた様子

```
augmentedTrainingData =  
transform(trainingData,@augmentData);
```

ネットワークの入力に
合わせて正規化

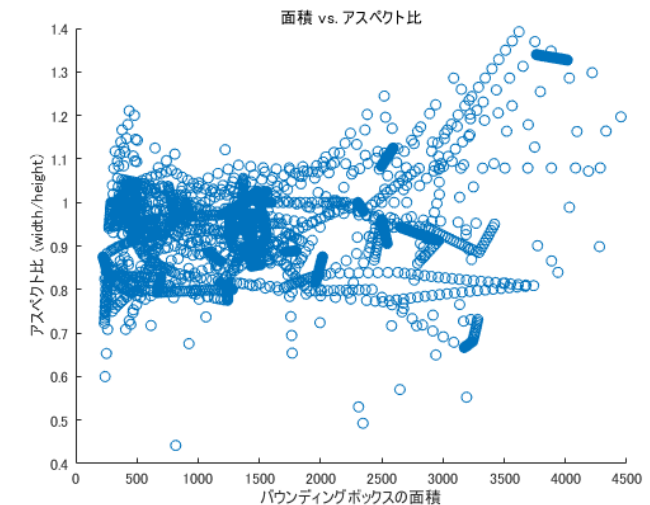
```
preprocessedTrainingData =...  
    transform(augmentedTrainingData,...  
    @(data) preprocessData(data,inputSize));
```

アンカーボックスの探索： クラスタリングでアンカーボックスを決定

```
numAnchors = 4;  
[anchorBoxes, meanIoU] = estimateAnchorBoxes(trainingDataForEstimation, numAnchors)
```

クラスタリングで得られた4つのアンカーボックス(**anchorBoxes**)

```
anchorBoxes = 4×2  
    123    46  
     53    19  
    100    37  
     26     4
```

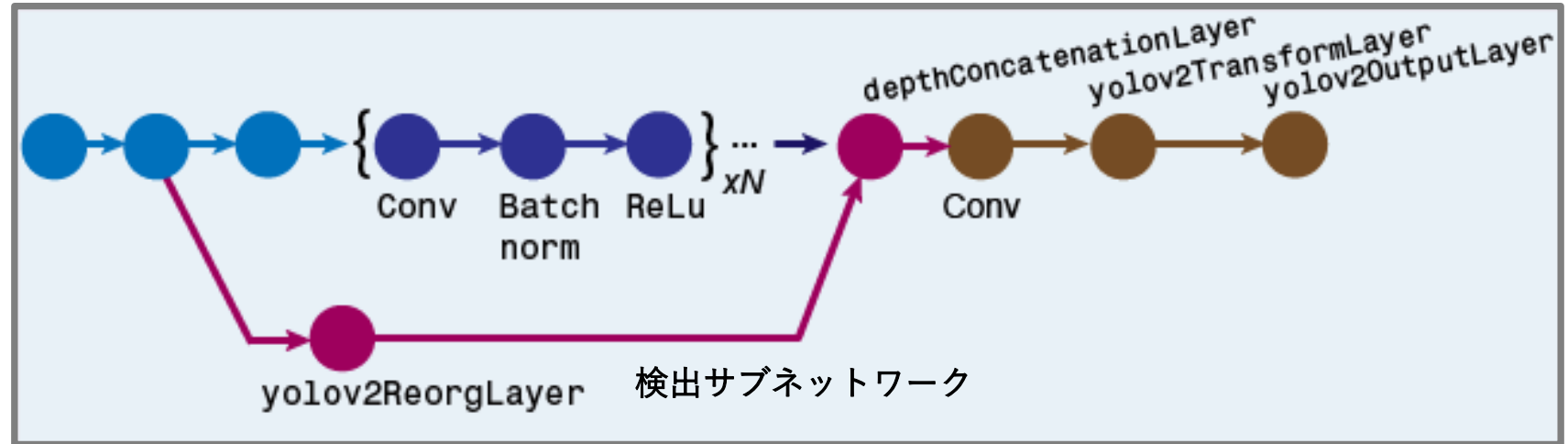


<https://www.mathworks.com/help/vision/ug/estimate-anchor-boxes-using-clustering.html>

物体検出器のネットワークの定義

Deep Learning Toolbox
Computer Vision Toolbox

学習済みモデル
特徴抽出器
(例: ResNet 18)



```
% 前段の特徴抽出に使う学習済みモデル
network = resnet18();
% 特徴抽出として使うレイヤーを指定
featureLayer = 'res5b_relu';
% 入力画像サイズ
imageSize = network.Layers(1).InputSize;
% クラス数
numClasses = width(trainingDataset)-1;
% YOLO v2物体検出ネットワークを定義
lgraph = yolov2Layers(imageSize, numClasses, ...
    round(anchorBoxes), network, featureLayer);
% 整合性確認
analyzeNetwork(lgraph);
```

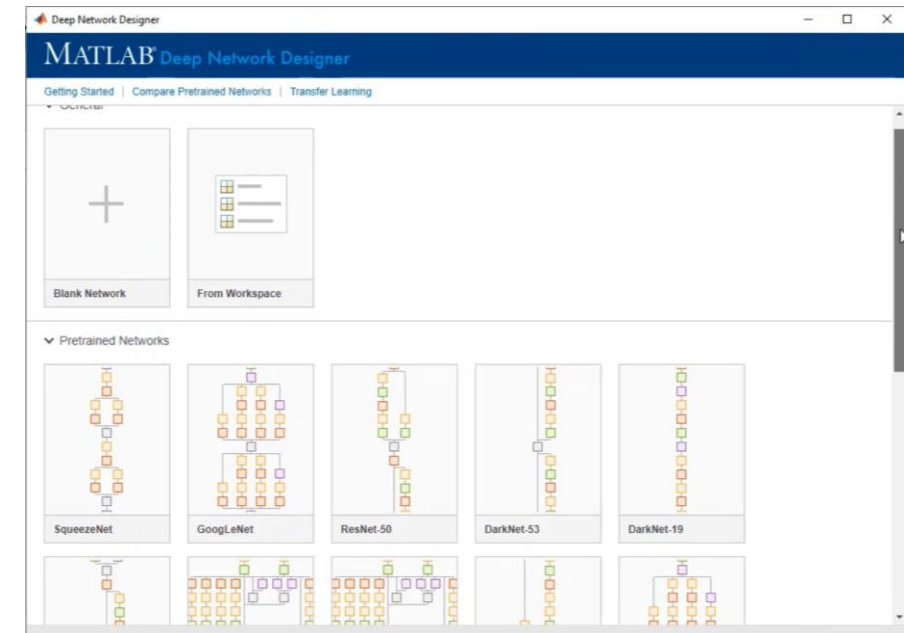


ネットワークを直感的に設計

ディープネットワークデザイナー を使ってインタラクティブにネットワークの構築、可視化、編集、学習

アプリを使って下記が可能:

- ネットワークのコーディングやテストを、手作業で行う必要なし
- ネットワークを編集したり、ゼロから新しいネットワークを構築
- ドラッグ&ドロップで新しいレイヤーを追加し、新しい接続を作成
- レイヤのプロパティを表示、編集



ネットワークを構築、可視化、編集するための
ディープネットワークアプリ

初期リリース	R2018b
MATLAB コード生成	R2019a
データの取り込みと学習	R2020a
学習済みネットワーク	
時系列のサポート	R2020b

実験で最適なネットワークを見つける

様々な初期条件でネットワークを学習する実験を行い、その結果を比較

下記に利用可能:

- ハイパーパラメータの値の範囲を探索する
- 異なるデータセットを使用した場合の結果を比較する
- 異なるディープネットワークアーキテクチャのテスト

実験マネージャーアプリ

- 実験をコード化したり、手作業での管理を低減

初期リリース

R2020a

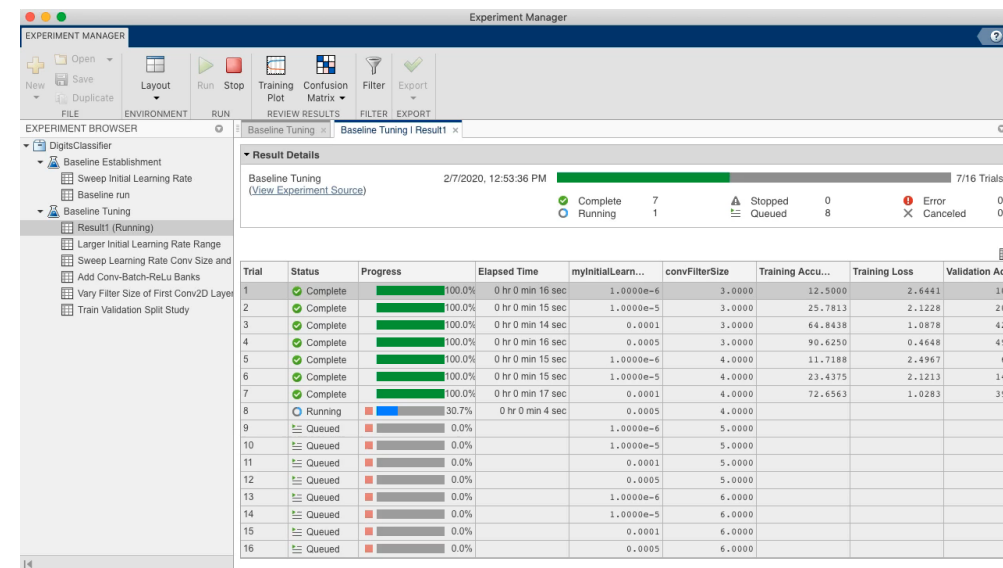
ネットワーク学習の並列化

R2020b

ベイズ最適化

カスタムトレーニンググループ

R2021a



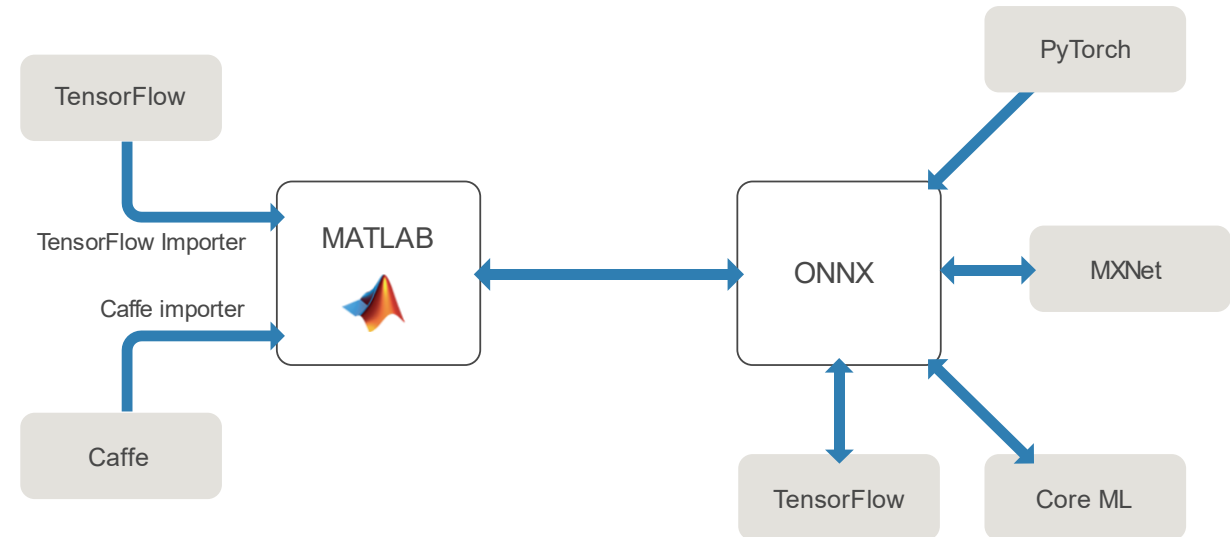
複数の条件での最適なネットワーク探索や解析、結果の比較を行う実験マネージャーアプリ

オープンソースフレームワークとの相互運用性

フレームワークの相互運用性 によりデータサイエンス、エンジニアリング、実装におけるギャップを埋める

モデルのインポート/エクスポートが可能:

TensorFlow-Keras インポーター	R2017b
ONNX コンバーター	R2018a
<u>TensorFlow モデルインポーター</u>	R2021a



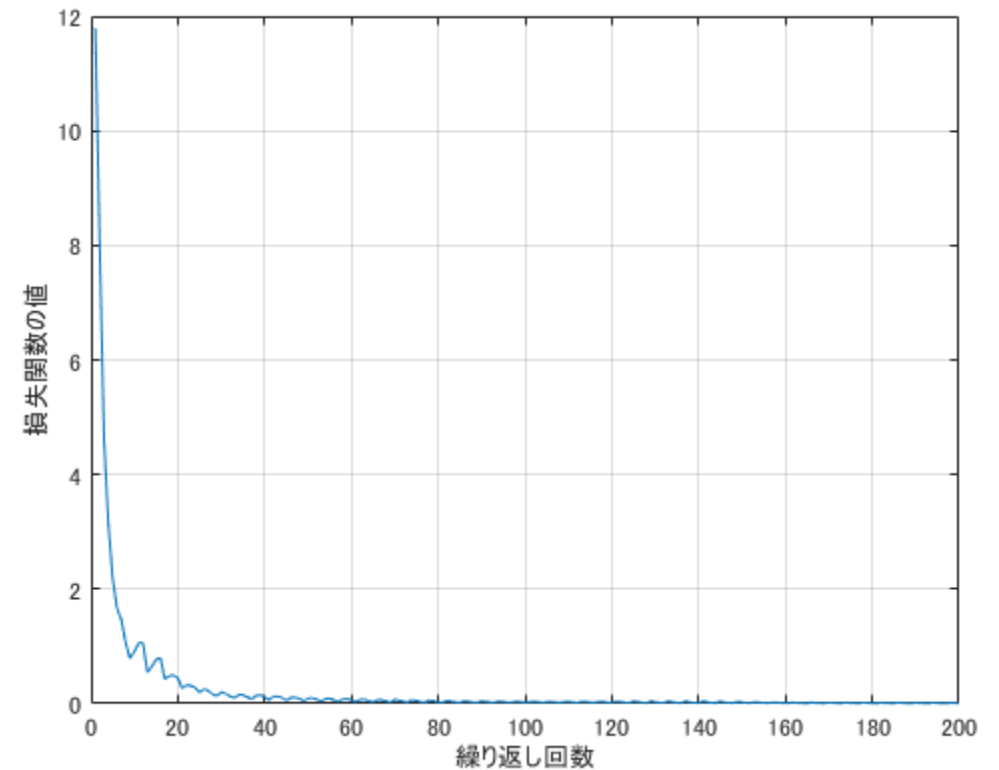
YOLO v2の学習

R2019aDeep Learning Toolbox
Computer Vision Toolbox

▪ `trainYOLOv2ObjectDetector`

- YOLO v2物体検出器の学習
- GPUを使用した高速化

```
>> [detector, info] =...  
trainYOLOv2ObjectDetector(trainData,lgraph,options);  
  
>> detector  
detector =  
  
yolov2ObjectDetector のプロパティ:  
  
    ModelName: 'left'  
      Network: [1×1 DAGNetwork]  
TrainingImageSize: [224 224]  
   AnchorBoxes: [4×2 double]  
   ClassNames: [left    right  stop]
```



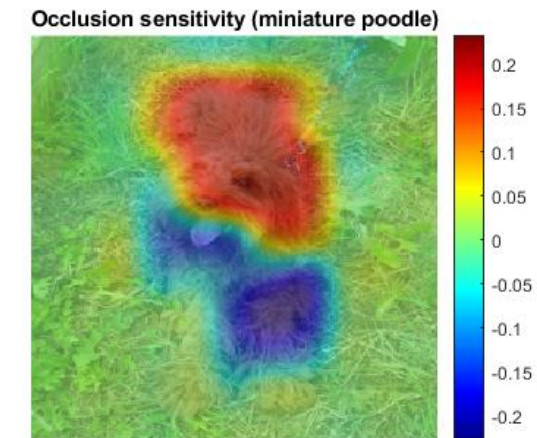
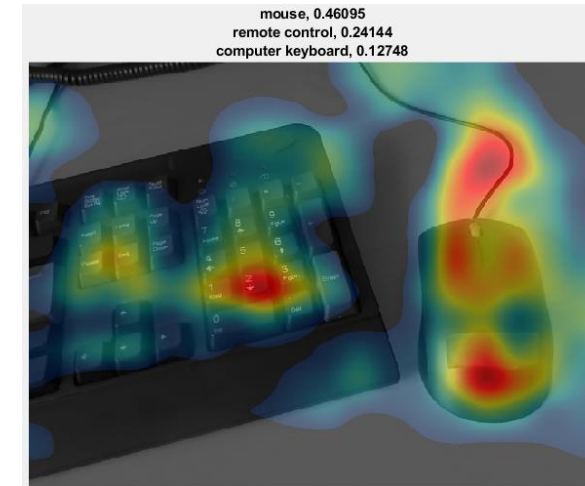
モデルの動作を説明し、可視化する

ネットワークの予測を調査し、説明するための可視化技術

それぞれの層の特徴出力の可視化

推論にどの特徴が使われているかを可視化

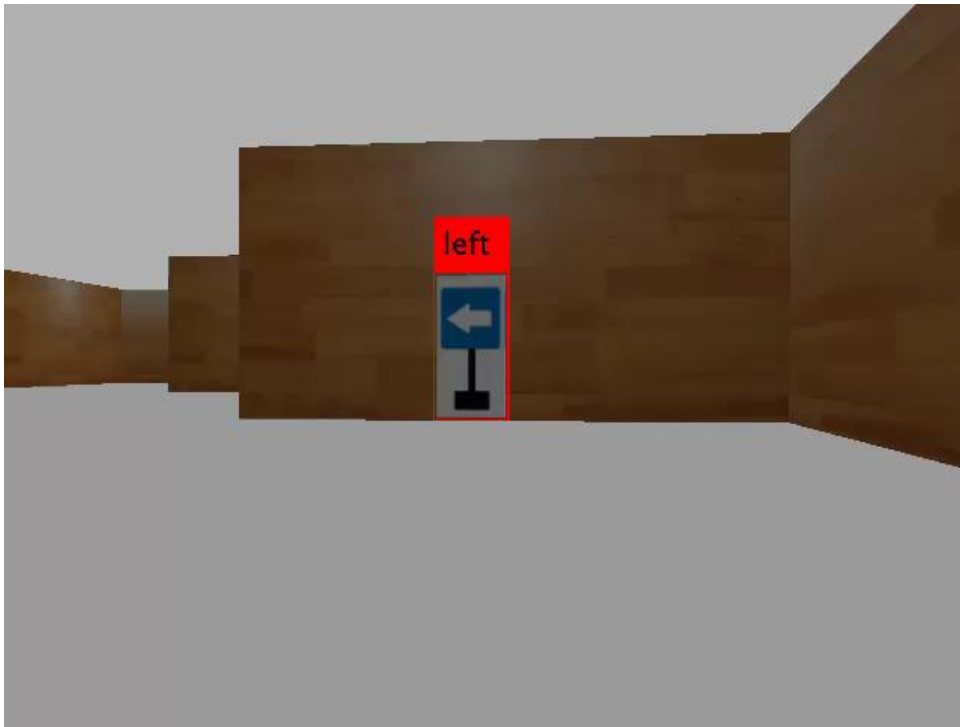
Deep Dream	R2017a
Class Activation Mapping	R2019a
Occlusion	R2019b
Local Interpretable Model-agnostic Explanation (LIME)	R2020b
GRAD-CAM	R2021a



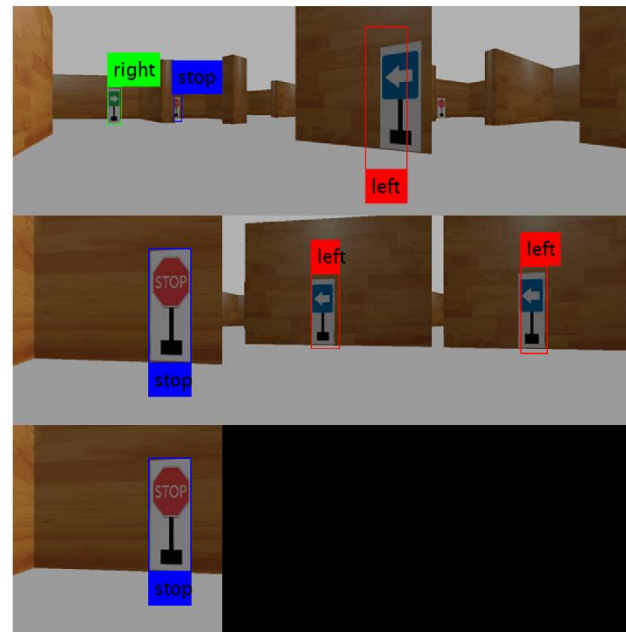
YOLO v2の学習モデルの評価

Deep Learning Toolbox
Computer Vision Toolbox

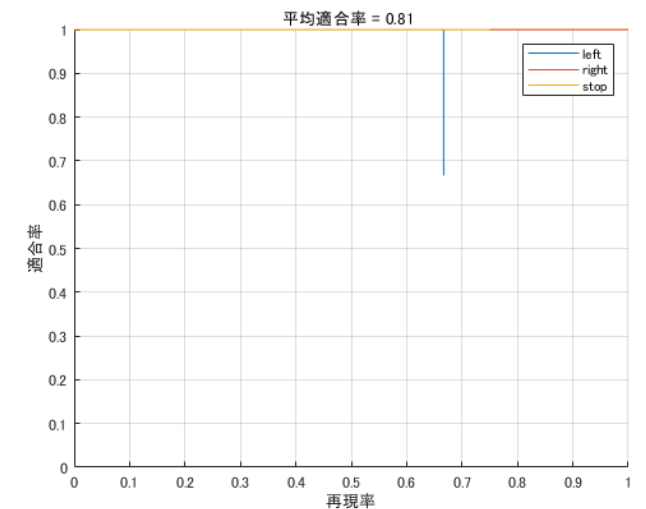
- 動画へ適用し目視確認



- テスト画像で定量評価



検出結果



PR曲線

システム全体でのディープラーニングを評価する

ディープラーニングモデルの影響を、制御、信号処理、センサーフュージョンの各コンポーネントでシミュレート可能

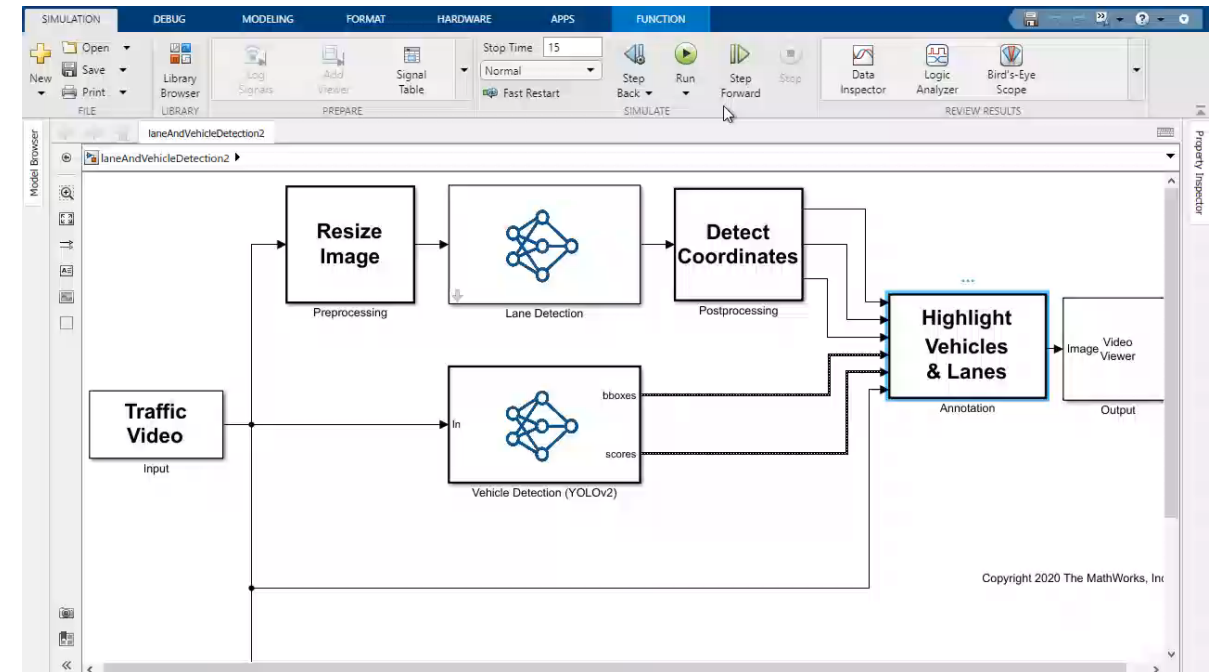
学習したモデルをシステムモデルのシミュレーションに含める

画像分類とネットワーク予測

R2020b

リカレント・ニューラル・ネットワーク

R2021a



ディープラーニング標識検出とライントレースの NVIDIA® JetBot 実装

ROS Toolbox
Deep Learning Toolbox™
Computer Vision Toolbox™
MATLAB Coder™
GPU Coder™

ROS接続

アルゴリズム
開発

Gazeboシミュ
レーション

実機実装



実機

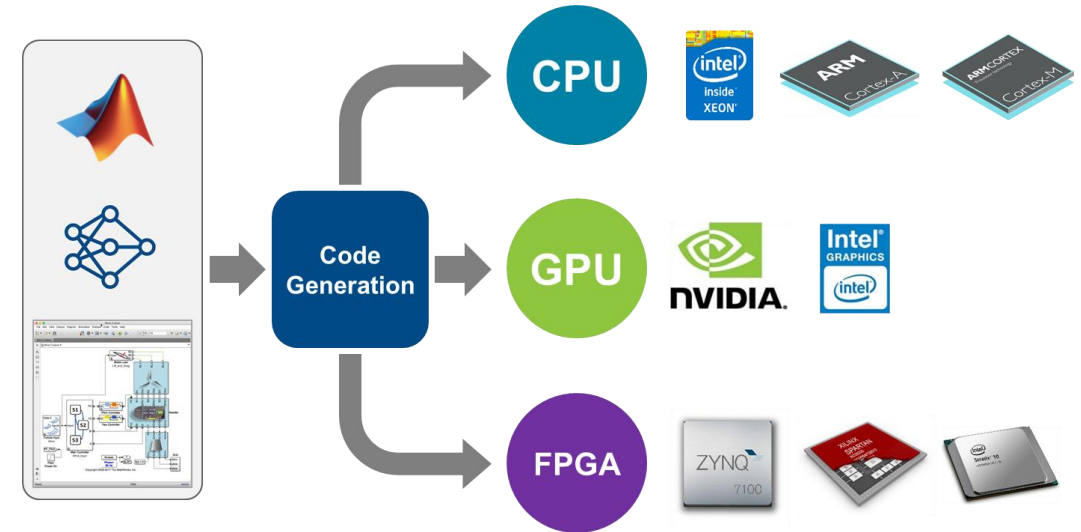
シミュレーション

組み込み機器のコードを自動的に生成する

学習したモデルを低コストの試作ボードでテストしたり、モデルを再コーディングせずに製品化

最適化されたネイティブコードを生成:

GPU – GPU Coder	R2017b
CPU – MATLAB Coder	R2018b
FPGA – Deep Learning HDL Coder	R2020b
標準化C/C++コード生成	R2021a



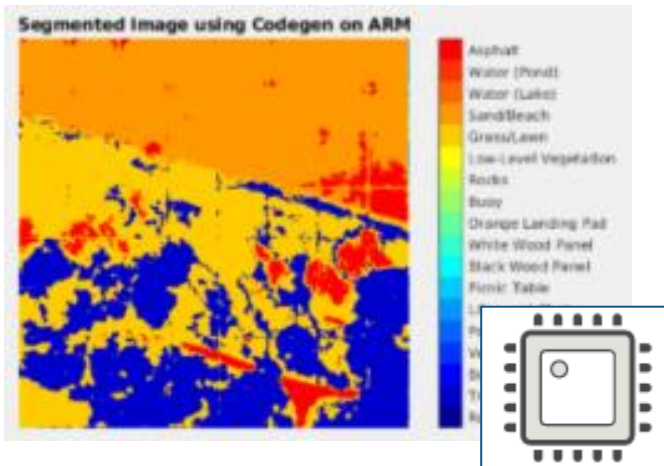
低コストのボードでモデルをテスト:

Raspberry Pi	R2019a
ARM Targets	



自動コード生成と実装

ARM Neon コード生成

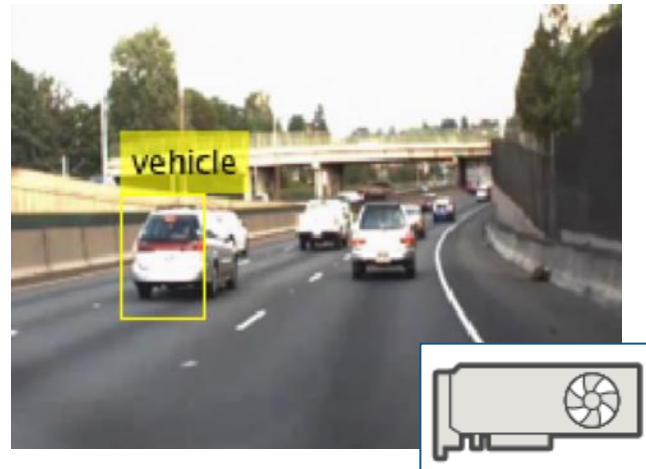


Code Generation for Semantic
Segmentation Application on
ARM Neon

Deep Learning Toolbox™
MATLAB Coder™

R2020a

NVIDIA GPU CUDAコード生成

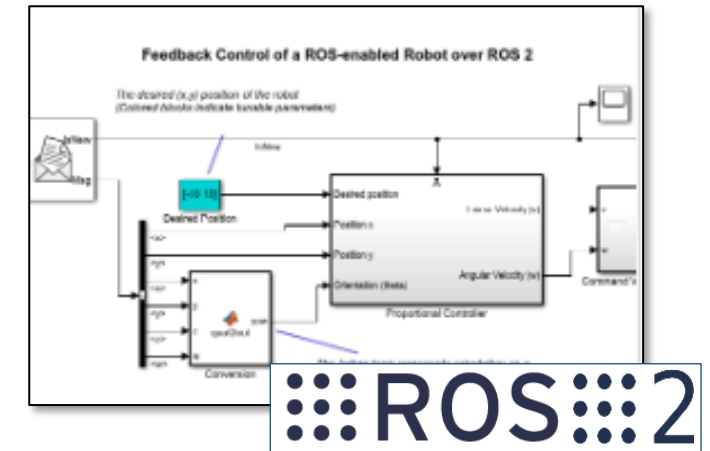


Code Generation for Object
Detection by Using Single
Shot Multibox Detector

Deep Learning Toolbox™
GPU Coder™

R2020a

ROS / ROS2 ノード生成



ROS2

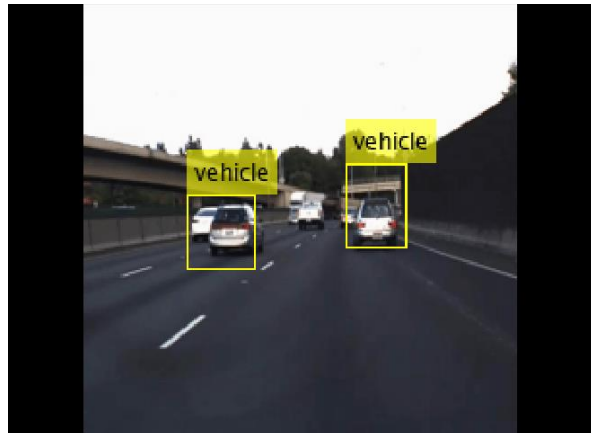
Generate a Standalone ROS 2
Node from Simulink®

ROS Toolbox
Embedded Coder™

R2020a

自動コード生成と実装

Intel MKL-DNN コード生成



YOLO v2 および Intel MKL-DNN を使用したオブジェクト検出の C++ コードの生成

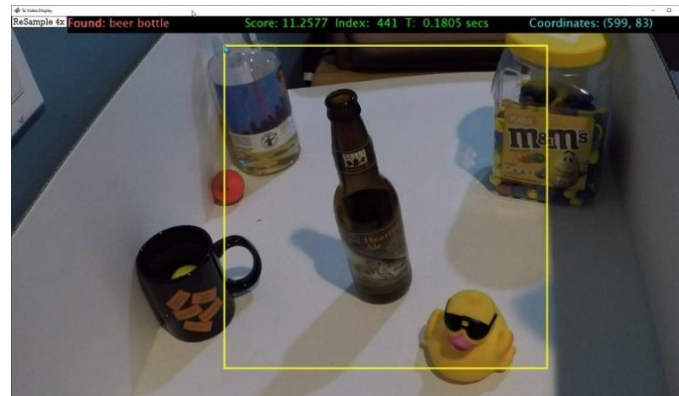
MATLAB Coder™

Deep Learning Toolbox™

Computer Vision Toolbox™

R2020b

FPGA



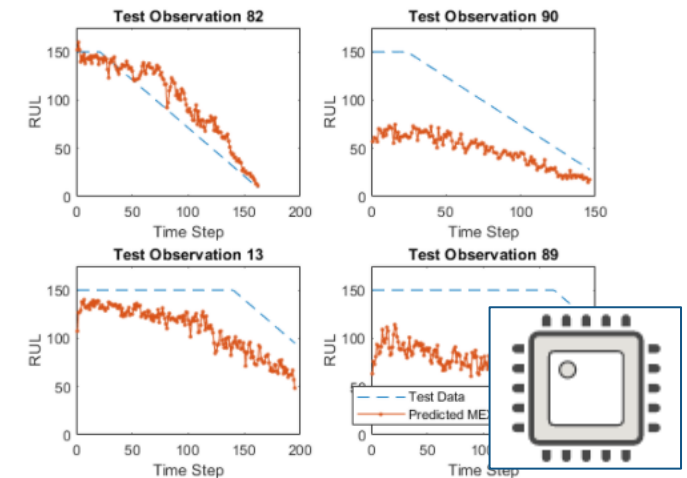
Run a Deep Learning Network on FPGA with Live Camera Input

Deep Learning HDL Toolbox™

HDL Coder™

R2020b

ディープラーニング ポータブルCコード生成



Generate Generic C/C++ Code for Sequence-to-Sequence Regression That Uses Deep Learning

Deep Learning Toolbox™

MATLAB Coder™

R2021a

まとめ：MATLABによるディープラーニング

ディープラーニングを効率的に学習し、システムに統合

データ収集



データの
クレンジングや
前処理



ドメインナレッジ



シミュレーションに
よるデータの水増し

AI モデリング



モデルの設計と
チューニング



ハードウェアによ
る学習の高速化



他フレームワーク
連携

システムデザイン



複合的なシステムへ
の統合



システム
シミュレーション



システムの検証

展開



組み込みデバイス



エンタープライズ
システム



エッジ, クラウド,
デスクトップ

画像のディープラーニング向けの製品構成

Toolbox		ディープラーニングにおける用途
Deep Learning Toolbox	必須	ディープニューラルネットワークの構築・学習
Image Processing Toolbox	ほぼ必須	静止画へのラベリング、前処理
Computer Vision Toolbox(※3)	ほぼ必須	動画へのラベリング RCNN(※1)/Fast-RCNN/Faster-RCNN/YOLOv2/SSD/YOLOv3の学習・推論 セマンティックセグメンテーション
Parallel Computing Toolbox	ほぼ必須	GPUによる学習・推論
Statistics and Machine Learning Toolbox	推奨	機械学習(※2)を使った転移学習 ベイズ最適化によるハイパーパラメータ調整
Signal Processing Toolbox	推奨	一次元信号の前処理 リサンプリング、欠損データの補間、簡単なフィルタリング
Audio Toolbox	推奨	音声信号の読み込み
MATLAB Coder GPU Coder	推奨	学習済みネットワークのCUDA C コード生成 (Intel® CPUs or ARM processors / NVIDIA GPU)
MATLAB Compiler / MATLAB Compiler SDK	推奨	ディープラーニングアプリケーションの展開 実行形式ファイル化、既存のWebシステムへの統合

※1 RCNNはStatistics and Machine Learning Toolboxも必要

※2 ディープラーニング以外

※3 Image Processing Toolboxが必要

※4 Parallel Computing ToolboxとMATLAB Coderが必要

ディープラーニングをこれからはじめる方向けの例題

Deep Network Designerで生成されたコードをExperiment Managerで使用するための適合	R2021a
イメージ分類用のデータストアの作成と確認	R2020b
ディープ ネットワーク デザイナーを使用したシンプルなイメージ分類ネットワークの作成	R2020b
ディープ ネットワーク デザイナーへのデータのインポート	R2020b
ディープ ネットワーク デザイナーでの image-to-image 回帰	R2020b
ディープ ネットワーク デザイナーでの簡単なセマンティック セグメンテーション ネットワークの作成	R2020b
ディープ ネットワーク デザイナーへのカスタム層のインポート	R2020b
事前学習済みのネットワークを使用した転移学習	R2020a
ディープ ネットワーク デザイナー入門	R2020a
転移学習入門	R2020a
SqueezeNetを用いた対話型転移学習	R2020a
シンプルなシーケンス分類ネットワークの作成	R2020a
最大・最小の活性化画像を用いた画像分類の可視化	R2020a

画像認識向けのディープラーニングの例題

Mask R-CNNディープラーニングを用いたインスタンスセグメンテーション	R2021a
ディープラーニングを用いたセグメンテーション・マップからの画像生成	R2020b
ディープラーニングを用いた体の姿勢の推定	R2020b
ディープラーニングによる映像とオプティカルフローデータからの活動認識	R2020b
YOLO v3 Deep Learningを用いた物体検出のためのコード生成	R2020b
PointNetによる点群分類	R2020a
YOLO v3ディープラーニングによる物体検出	R2020a
SSDによる物体検出	R2020a
学習済みONNX YOLO v2物体検出器のインポート	R2020a
YOLO v2 Object DetectorをONNXにエクスポート	R2020a
ディープラーニングを用いた物体検出の入門	R2019b
物体検出のためのバウンディング・ボックスの拡張	R2019b
セマンティック・セグメンテーションのためのピクセル・ラベルの拡張	R2019b
YOLO v2ディープラーニングによる物体検出	R2019a
深層学習を用いた3次元脳腫瘍のセグメンテーション	R2019a

オンライン無料学習

誰でも利用できる 11 時間分の
無料コンテンツ

- **ブラウザ上**で動かせます。
- いつでも開始、終了、再開
- ステップバイステップ
- 勉強したいところだけでも大丈夫
- **ゲーム感覚**で取り組みやすい

無料

MATLAB 入門 (日本語)

最短でMATLAB の基礎を学びましょう。

[コースを開始](#) [コース詳細](#)

無料

Simulink 入門 (日本語)

最短でSimulinkの基礎を学びましょう。本コースはSimulinkをインストールすると受講できます。

[詳細を確認する](#)

無料

機械学習 入門(日本語)

分類問題のための実用的な機械学習手法の基礎を学びます。

[コースを開始](#) [コース詳細](#)

無料

ディープラーニング入門 (日本語)

ディープラーニング手法を使用した画像認識を行う方法を学びましょう

[コースを開始](#) [コース詳細](#)

NEW 無料

画像処理入門 (英語)

MATLAB で実用的な画像処理の基本を学びます。

[コースを開始](#) [コース詳細](#)

無料

Stateflow 入門 (日本語)

Stateflow でステートマシンを作成、編集、およびシミュレーションするための基礎を学びます。

[詳細を確認する](#)

NEW 無料

Simulink による制御設計入門 (英語)

Simulink で基礎的なフィードバック制御系の設計方法を学びます。

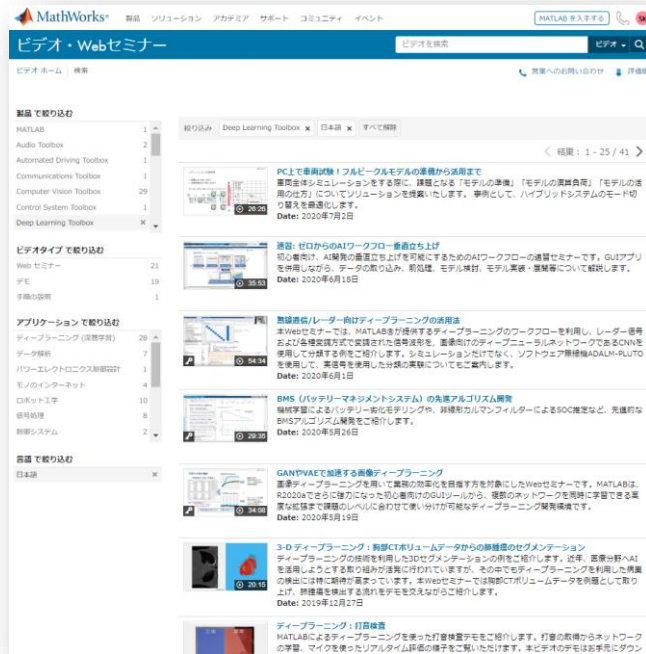
[詳細を確認する](#)



豊富な学習用コンテンツ

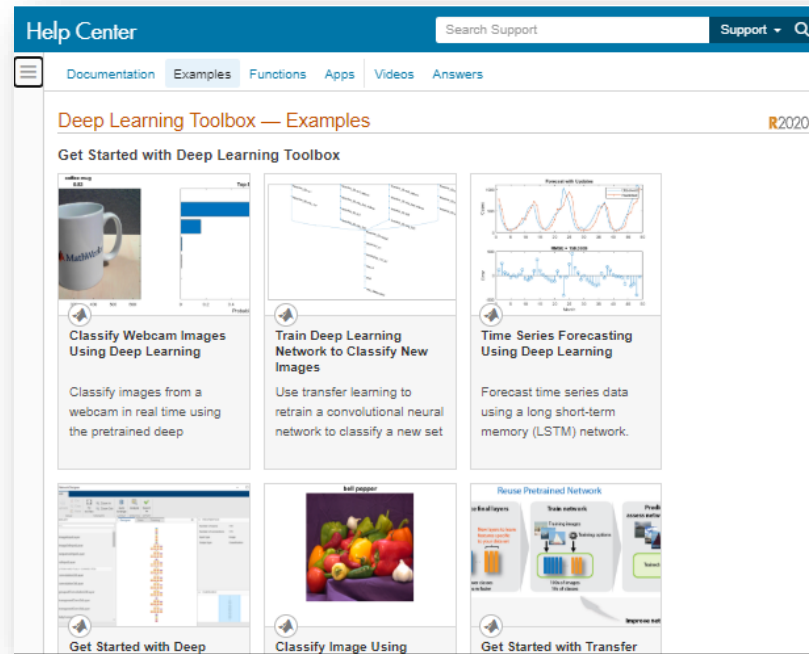
ビデオ/ウェビナー集

計12時間以上の動画教材



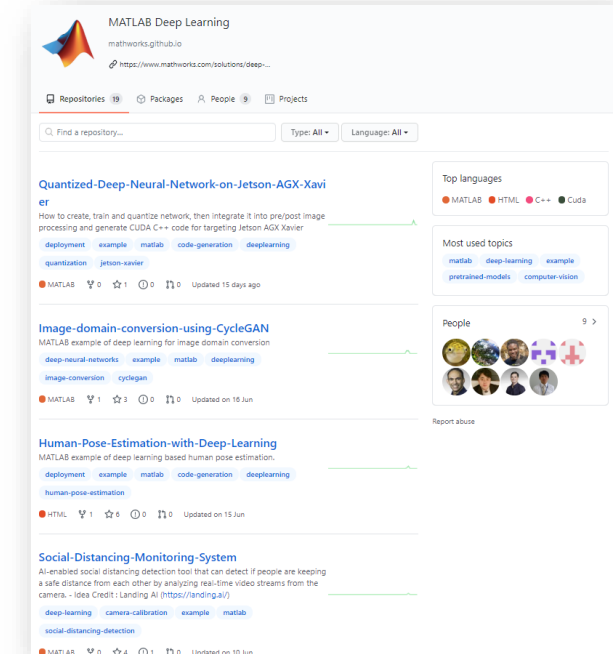
ディープラーニングサンプルコード

プロトタイピングに最適



GitHub リポジトリ MATLAB Deep Learning

最新のサンプルも！



MathWorksが提供するサポート



Training



Guided Evaluations



Onsite Workshops



Consulting



Technical Support



© 2021 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.