



筑波大学 知能ロボット研究室
University of Tsukuba, Intelligent Robot Laboratory

Deep Learningによる画像処理の実際

筑波大学大学院
知能ロボット研究室

小西 裕一 (Yuichi KONISHI)

2017/5/20 第15回ROBO-ONE Conference

自己紹介

□ 講演経緯

- 自律移動ロボットコンテスト「Argos Challenge」に認識担当として参加

□ 経歴

- 豊田高専(情報工学科)→筑波大学3編(情報科学類)→筑波大学大学院(システム情報工学研究科コンピュータサイエンス専攻)

□ 知能ロボット研究室

- 自律移動ロボットの研究
- ハードウェアからソフトウェアまで

□ 研究

- 自律移動ロボット上での深層学習による画像認識
 - ArgosChallenge,つくばチャレンジの画像認識担当



研究室メンバー

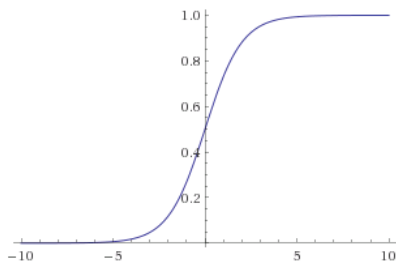
- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

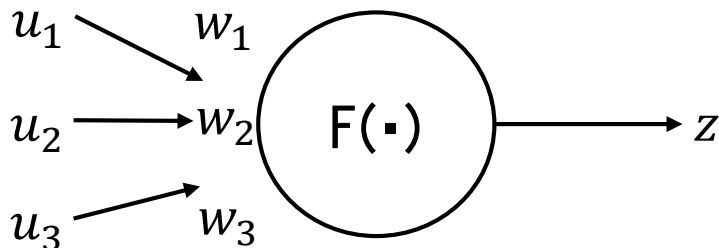
深層学習とは

- ニューラルネットワークを多層化した機械学習手法
 - 画像、音声、文字列などに応用される

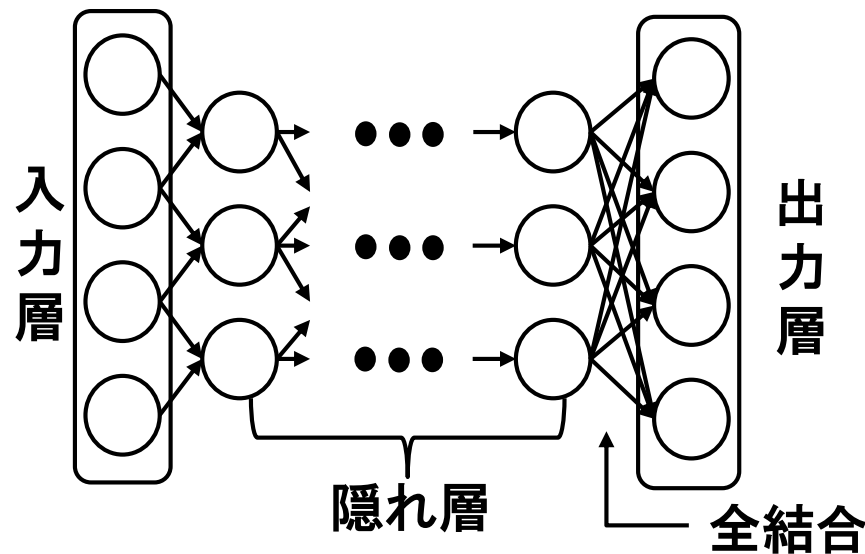
活性化関数
 f : シグモイド
 関数など



$$z = f \left(\sum u_i * w_i \right)$$



ニューロンモデル



多層ニューラルネットワーク

w を調整することで任意
 の関数を近似できる

画像分野における深層学習

□ CNN(Convolutional Neural Network)

- 畳み込みニューラルネットワーク
- **畳み込み層**、活性化関数、プーリング層、全結合層からなる

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

入力画像例

0	0	0
1	1	1
0	0	0

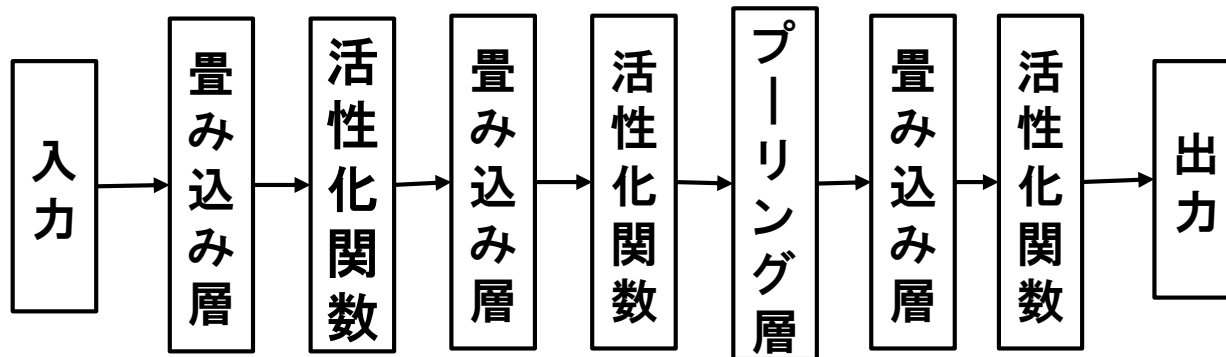
重み $W(3 \times 3)$ の例



1	1
1	1

次元圧縮される

画像の特徴が
つまっている

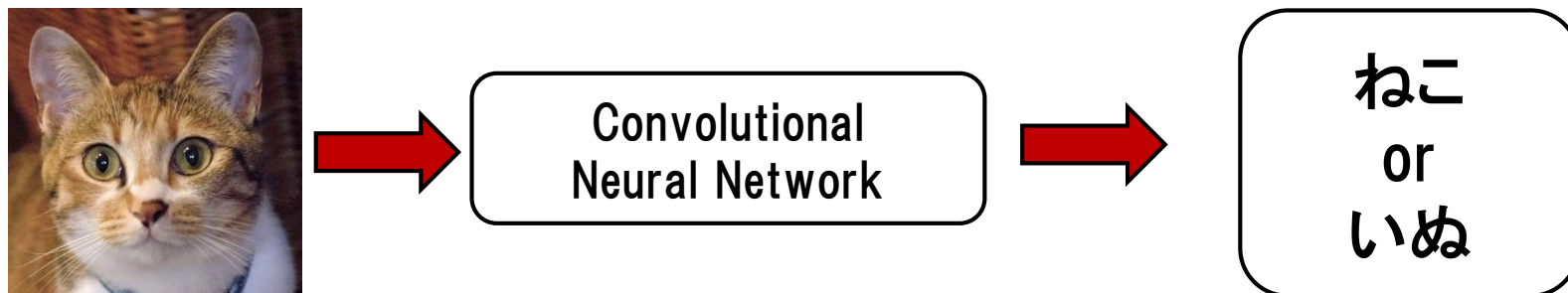


畳み込みニューラルネットワークの構成例

深層学習で解く代表的な問題

□ 分類問題

- 入力画像がどのクラスに属するか分類する問題(確立出力)



2クラス分類の例(多クラスでも良い)

□ 回帰問題

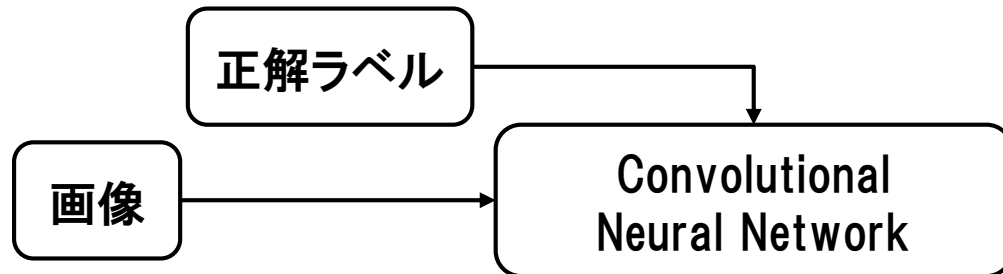
- 入力画像に対して、実数値を出力する問題



4つの実数出力の例

深層学習における学習と推論

□ 学習フェーズ



, 0



, 1

学習データの例(分類)

- 大量の学習データ(画像と正解ラベルのペア)より、ニューラルネットワーク内部のパラメータを調整して学習
 - 正解ラベルとの誤差を元にパラメータを調整

□ 推論フェーズ

- 学習データに含まれない画像を与える

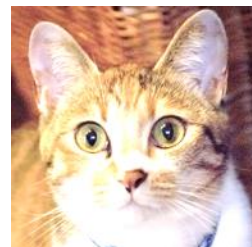
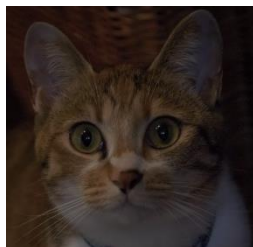


ねこ:0
or
いぬ:1

推論の例(2クラス分類)

学習データについて

- 一般的には撮影した画像を用いる
 - 認識したい物体を様々な条件で大量に撮影
 - 十分な認識性能を得るには数万枚以上の画像が必要
 - 正解ラベルをそれぞれの画像に割り当てる
 - **データ拡張**
 - 画像処理によって想定される様々な環境の画像を作り出し、学習データに用いる
- より環境の変化に強い認識が期待できる
- 物体の大きさ、位置、明るさなど

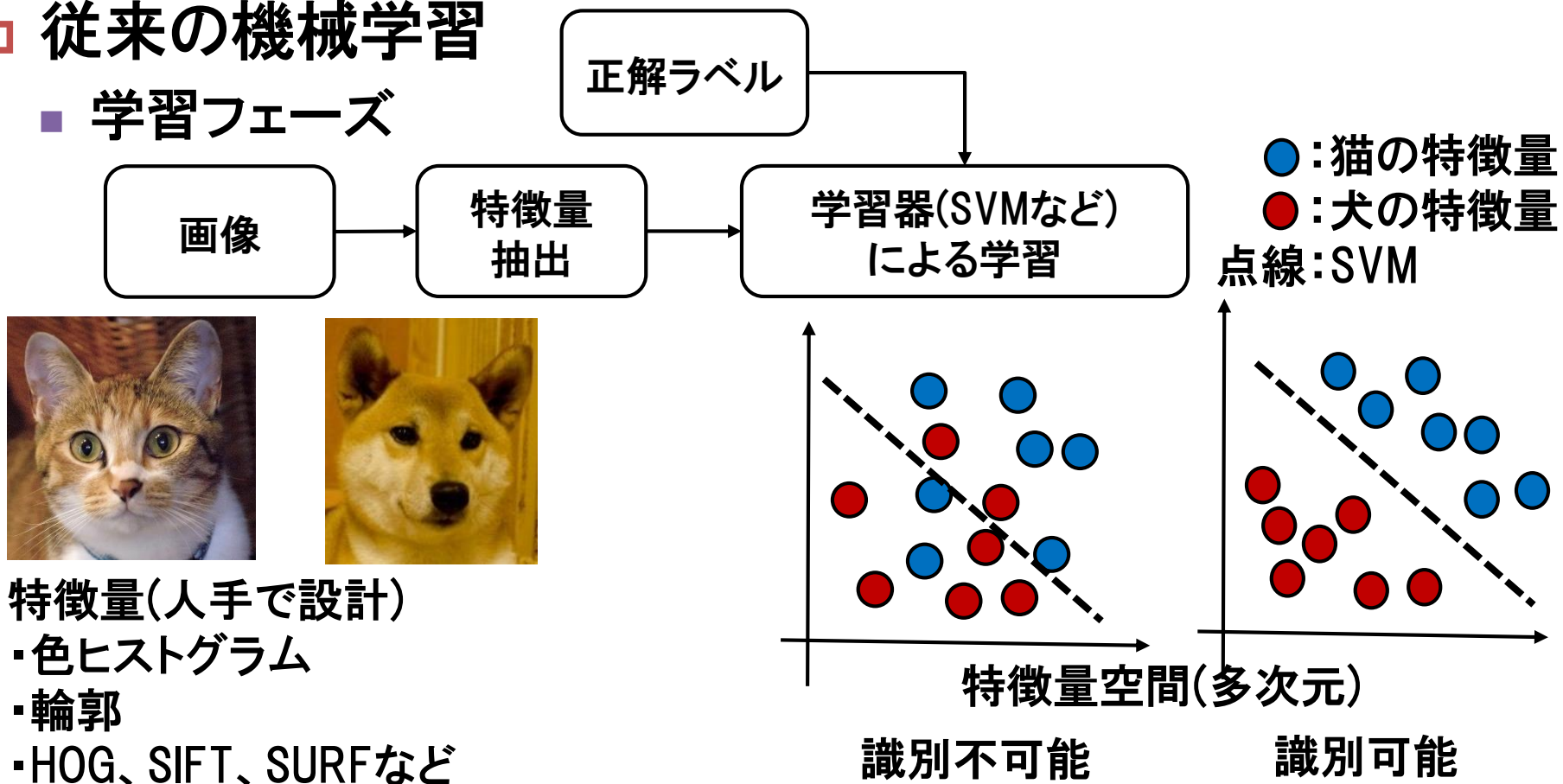


画像処理による明るさ変換

深層学習と従来の機械学習の違い

□ 従来の機械学習

■ 学習フェーズ



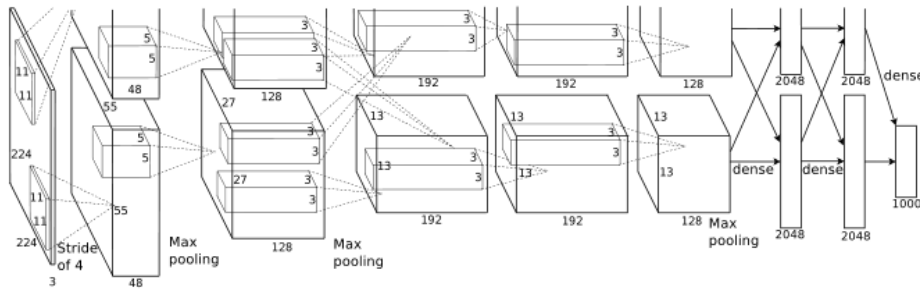
特徴量の決め方が重要とされていた

→ 深層学習では特徴量を決める必要がない

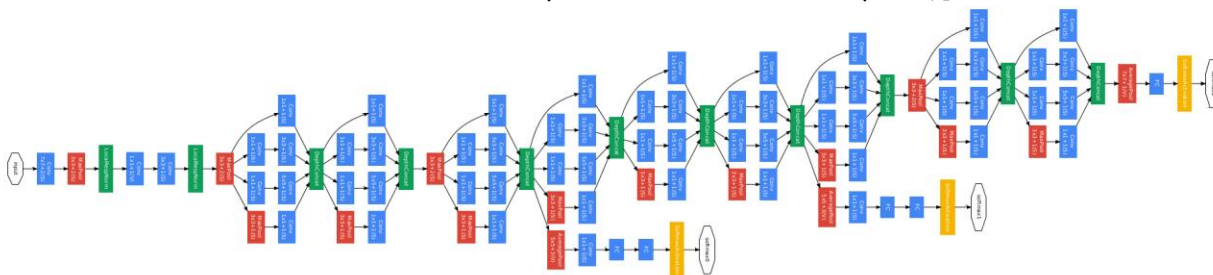
- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

代表的なCNN(1/2)

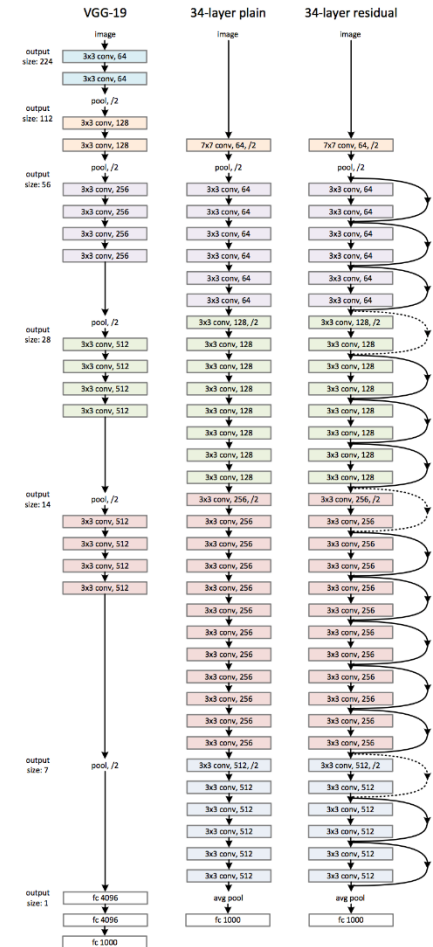
- 分類問題用ネットワーク(回帰問題にも使える)
- ILVRC
 - ImageNet Large-scale Visual Recognition Challenge
 - 画像を1000クラスに分類



AlexNet (ILSVRC 2012) 8層



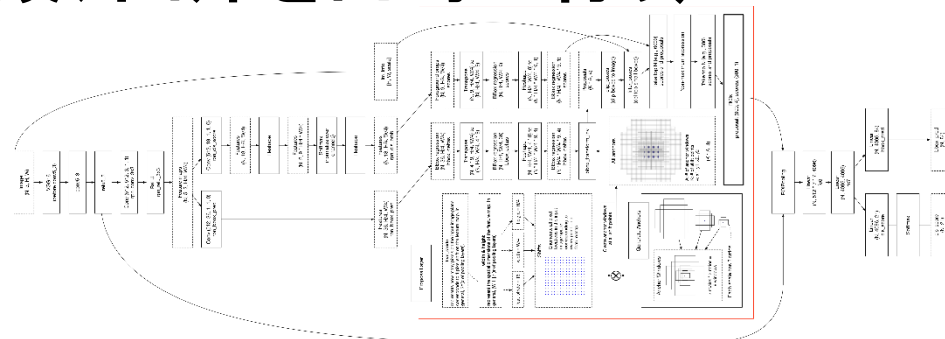
GoogLeNet (ILSVRC 2014) 22層



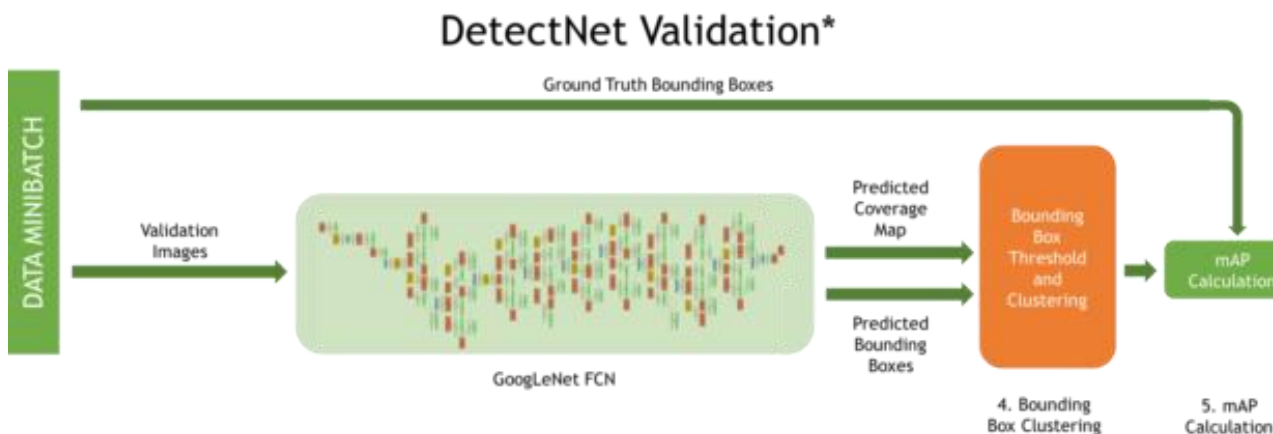
ResNet(ILSVRC 2015) 152層

代表的なCNN(2/2)

- 物体検出ネットワーク(分類、回帰を同時に行う)
- Faster-RCNN(2015)
 - S,REN氏によって開発
- DetectNet
 - Nvidia社によって開発(2016)



Faster-RCNNの全体図



DetectNetの全体図



物体検出例

出典: <https://devblogs.nvidia.com/parallelforall/detectnet-deep-neural-network-object-detection-digits/>

深層学習のフレームワークの紹介

□ Caffe

- カリフォルニア大学バークレー校が中心
- C++で実装、Python, MATLABでも使える
- 定義ファイルよりネットワークを定義
- 少しセットアップが大変
- 先月Caffe2(NVIDIA+Facebook)が開発
- 学習、推論ができる

```

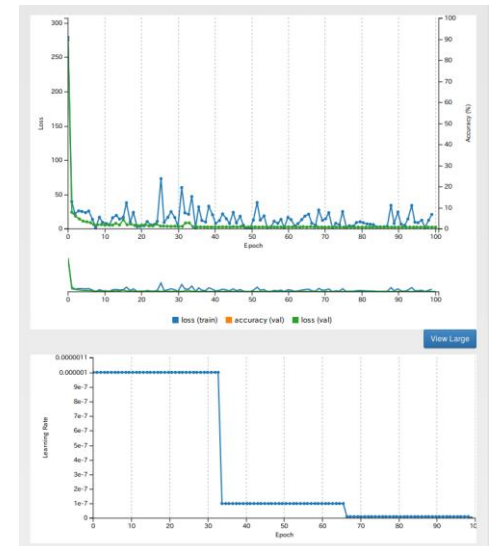
1 # AlexNet
2 name: "AlexNet"
3 layer {
4   name: "conv1"
5   type: "Convolution"
6   bottom: "data"
7   top: "conv1"
8   param {
9     lr_mult: 1
10  }
11  decay_mult: 1
12 }
13 param {
14   lr_mult: 2
15   decay_mult: 0
16 }
17 convolution_param {
18   num_output: 96
19   kernel_size: 11
20   stride: 4
21   weight_filler {
22     type: "gaussian"
23     std: 0.01
24   }
25   bias_filler {
26     type: "constant"
27     value: 0
28   }
29 }
30 layer {
31   name: "relu1"
32   type: "ReLU"
33   bottom: "conv1"
34   top: "conv1"
35 }
36 layer {
37   name: "norm1"
38   type: "LRN"
39   bottom: "conv1"
40   top: "norm1"
41   lr_param {
42     local_size: 5
43     alpha: 0.0001
44     beta: 0.75
45   }
46 }

```

Caffeによる定義ファイル

□ DIITS

- Caffeを使いやすくするツール(NVIDIA社)
- ブラウザを通じて、GUIで操作可能
- 学習過程をグラフで描画
- 主に学習に使う



DISITによる学習過程の描画

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

自律移動ロボットにおける深層学習の使い方

□ 学習

- 計算コストが非常に高い
- GPUを積んだハイスペックパソコンで行う
→学習済みモデル(調整したパラメータを保存)を生成

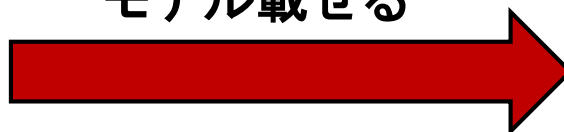
□ 自律移動ロボット上で推論

- 学習済みモデルを読み込む
- 学習に比べると推論の計算コストは低い



学習済み
モデルを生成

事前に学習済み
モデル載せる



推論を行う

自律移動ロボットによる認識の難しい点

- 認識対象となる物体をきれいに撮影できない
 - 対象物体が正面から撮影できるとは限らない
 - 画像がぶれる、対象物の一部が隠れることがある
 - 屋外の場合、照明条件が変化しやすい

→環境の変化に強い物体認識が必要

→学習データが重要、データ拡張
- ロボットに搭載できるコンピュータの性能に限りがある
 - コンピュータ大きいとロボットに載せれない
 - 消費電量の大きいコンピュータだとバッテリーがもたない
 - 認識が十分な早さで計算できるかわからない

→計算時間を意識した物体認識が必要

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

18 ARGOS Challengeについて

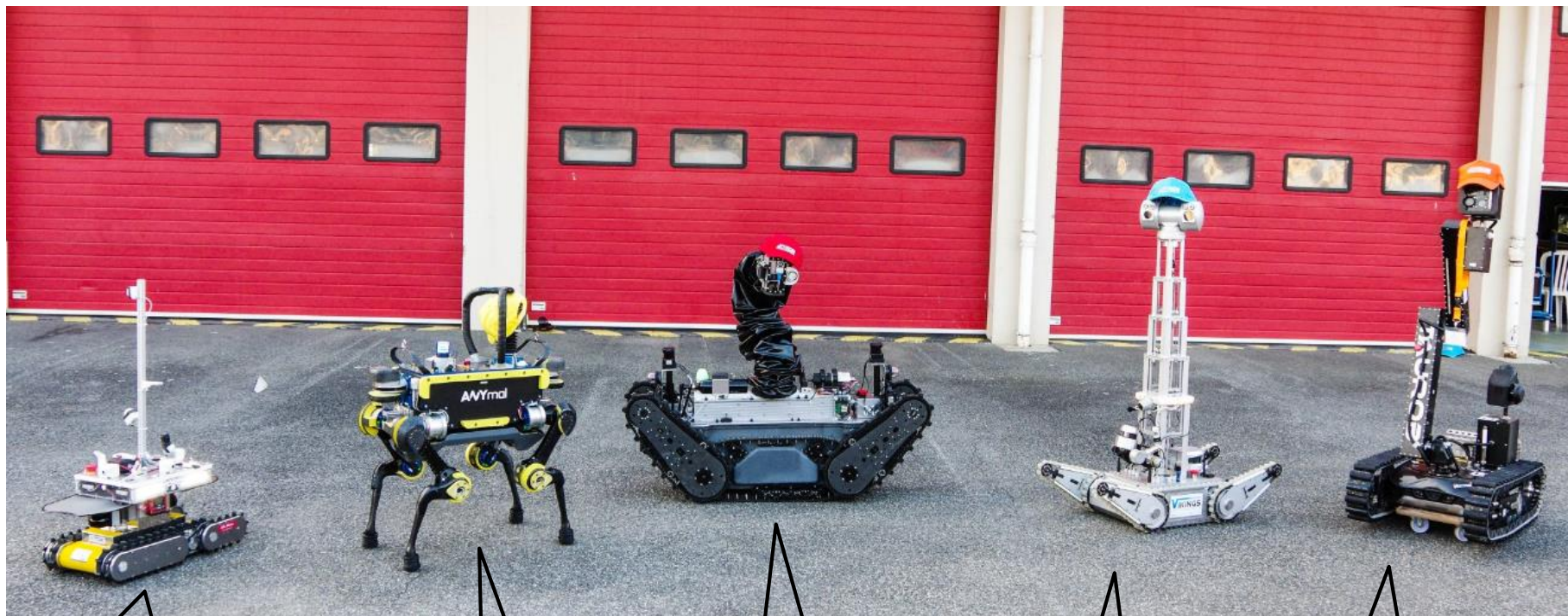
- ARGOS(Autonomous Robot for Gas and Oil Site)
 - 仏国の総合石油企業TOTALと仏国立研究機構が共催
 - 計器の自動点検(メータ、バルブ、水位計)、ガス漏れや火災などを検出するロボットの開発競争(世界5チーム参加)
 - 東北大の永谷先生の依頼より、画像処理に関して筑波大学が共同研究として参加



コンテスト用模擬環境

模擬環境の内部

参加チーム



日本: AIR-K
認識について:
カメラ6つ搭載
JetsonTX1搭載

スイス
L10

スペイン&
ポルトガル
FOXIRIS

フランス
VIKINGS

オーストリア
&ドイツ
ARGONAUTS

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - **メータ読み取り**
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

メーター検出、読み取りの流れ

ロボットを
対象物へ移動



自律移動ロボット

メータ画像撮影



自律移動ロボットのカメラから撮影した画像

メータ検出



画像中から切り出した
メータ画像

メータ読み取り

Deep Learning
を用いる

メータの針の角
度を入力

メータ読み取り

- メータの指針の角度を読み取る
 - 現場のメータと照らし合わせることで値がわかる
- ロボット移動後、認識をすれば良い
 - 計算時間をかけても良い
- メータ読み取りの難しい点
 - メータを正面から撮影できるとは限らない
 - メータ撮影時の環境の変化
 - メータのガラスの光の反射
 - 照明条件の変化



メータを拡大させた画像



ロボットより撮影される画像の例

メータ認識用いる学習データ(1/2)

- データ拡張によって学習用メータ画像を生成する
 - 正面から撮影した元画像5枚
- 指針をランダムに回転させて盤面画像に合成
 - 指針と盤面画像に分ける(人手)
 - 盤面の直径に対する指針の浮き具合の比率を元に、3次元的に指針を浮かせて配置、さらに指針を回転
 - 投影変換を行い正面以外の視点(30度以内)から見た様子を再現



生成に使用した元画像の例



切り分けた画像例



合成したメータ画像

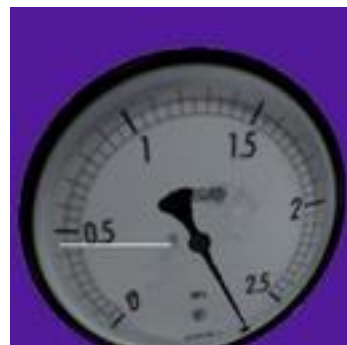
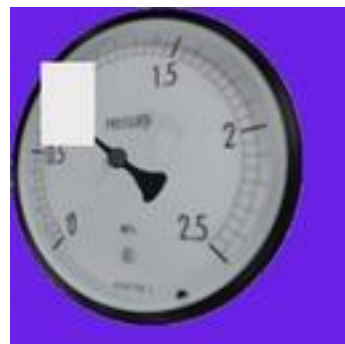
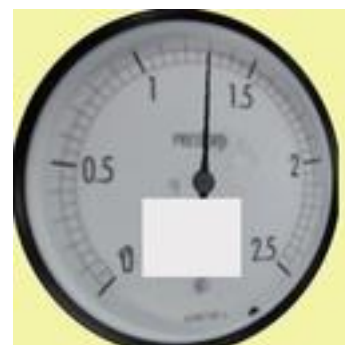
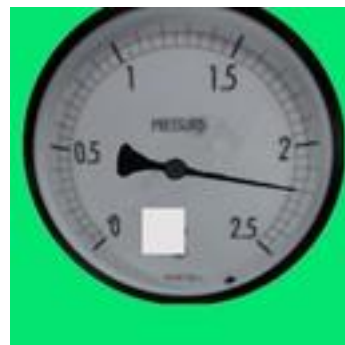
学習用メータ画像の生成(2/2)

- 合成した画像に以下の画像処理をランダムなパラメータで行うことで想定される環境を表現
 - γ 補正による明るさの変化
 - γ 値 0.3~2.8(0.1刻み)
 - メータの位置の移動、メータの大きさの変化
 - 位置 -10%~10% (0.1%刻み)
 - 大きさ 90%~110% (0.1%刻み)
 - 白い四角の合成
 - 大きさ 0~35% (0.1%刻み)
 - 位置 0~90% (0.1%刻み)
 - 背景を単色で塗る(色)
 - RGB値
 - 平滑化(平滑化の度合い)



生成した学習用メータ画像の例

生成した学習用メータ画像の例



提案するメータ検出、読み取り手法

□ メータ検出

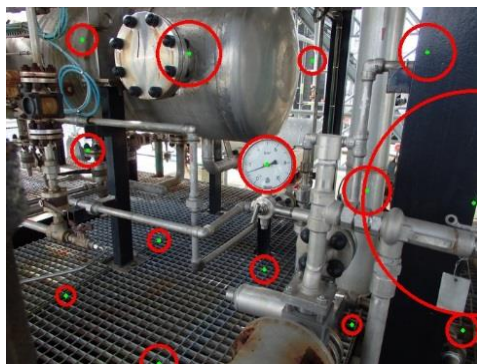
- ハフ変換(円の検出)による候補領域列挙 + CNNによる分類
- メータとメータ以外(背景)の2クラス分類

□ メータ読み取り

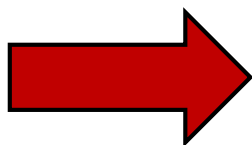
- CNNに切り出された画像を入力し、指針の角度を出力

→生成した学習用メータ画像を用いて2つのCNNのモデルを生成する

検出した円(メータの候補領域)に対してCNNによる分類

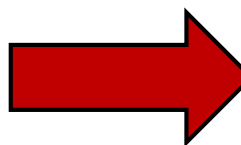


ハフ変換による円の検出



メータと判定された領域の切り出し

CNNによる指針の角度推定



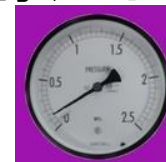
指針の角度を出力

□ メータ検出に使用するモデル

- 学習用メータ画像20,000枚(128x128)とメータ以外の画像(背景画像を切り出した)14,000枚の2クラスで学習
- ネットワークにはAlexNetを使用

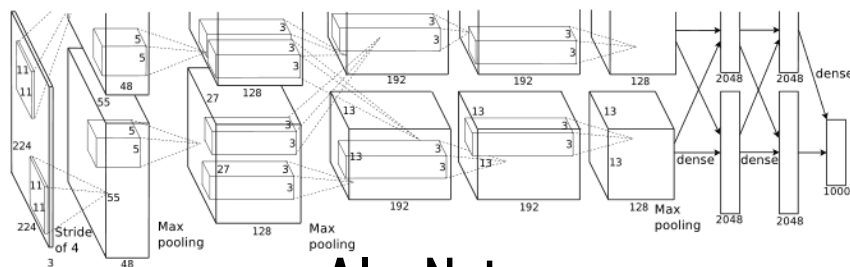
□ メータ読み取りに使用するモデル

- 学習用メータ画像20万枚(128x128)に角度を割り当て学習
- AlexNetで1変数を出力するように学習
- 指針の角度はメータの下を0度とし時計回り



(, 50)

読み取りの学習データ例



AlexNet

使用したツール

Nvidia社DIGITS(学習)
Caffe(モデル読み込み、予測)

使用したPC

- 学習に使用したデスクトップPC
 - HP Z840 Workstation
 - CPU:インテル(R) Xeon E5-260v3(2.40GHz,6コア)
 - メモリ:32GB
 - GPU:Quadro k5200×2枚
- 認識に使用した自律移動ロボット上のPC
 - JetsonTX1
 - 64ビットARM A57CPU
 - Maxwell 256-core GPU

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - **バルブ読み取り**
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

Tsukuba Challengeとは

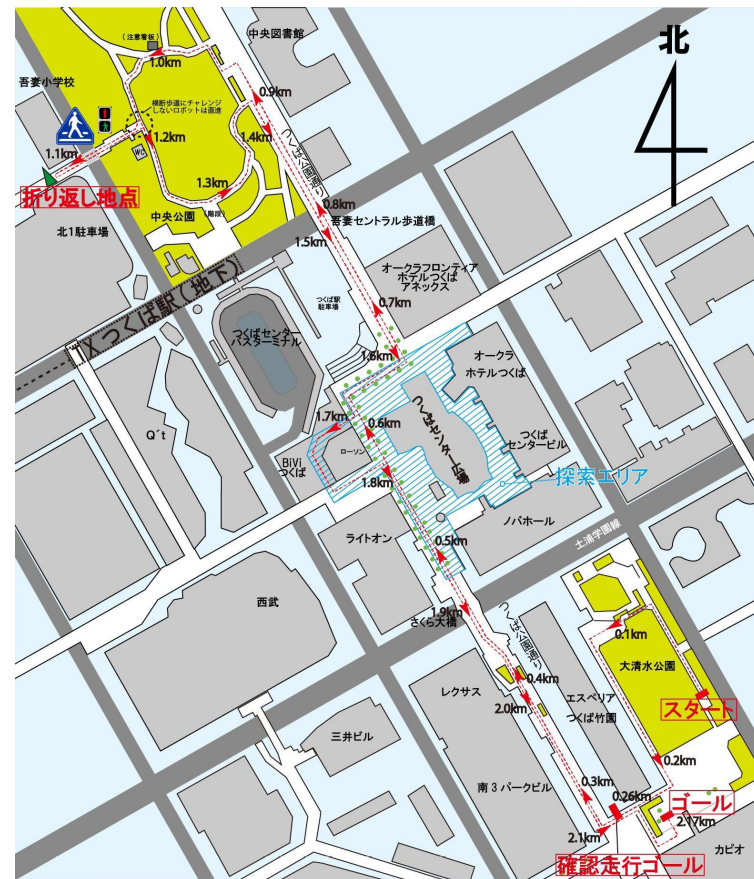
- つくば駅周辺の実環境で自律移動ロボットを走行させる技術チャレンジ
- 2016年の課題
 - 2,177mのコースを自律走行
 - 人物探索(4人)+横断歩道走行



探索対象の例

- 全国の大学、企業が参加

- 2016年は62チーム参加(筑波大3チーム) 2016年のコース



出典: <http://www.tsukubachallenge.jp/tc2016>

筑波大学チーム紹介

探索対象を
1人認識
本発表の人物認識



Rossy

参加チーム
唯一の
全課題クリア

本発表の信号
機認識が搭載



Kenaf

探索対象を
1人認識



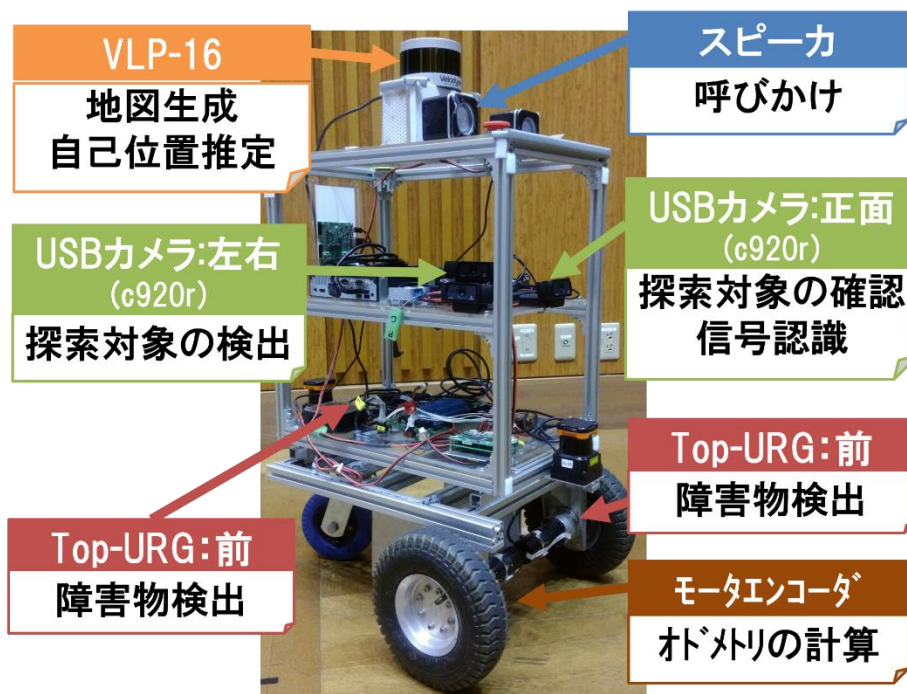
Kerberos

41 チームKenafの紹介

□ 特徴

- 人物認識、信号機認識を深層学習で行う
- 認識にCPU(Core i7 6700)を使用

ロボットのサイズ: W510×L620×H960 [mm]



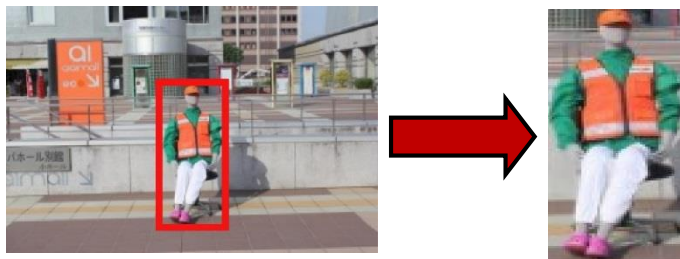
外装取り付け後の外観



Kenafのハードウェア構成

42 認識のコンセプトと制約条件

□ 物体検出を行った後、分類を行う



人物認識の例



信号機認識の例

□ 制約計算時間が異なる

■ 人物探索認識

- ロボットが移動する中での認識が必要
- リアルタイムに近い認識速度が必要

■ 信号機認識

- ロボットを停止させて認識すれば良い
- 信号機検出に時間をかけて良い
- 信号機の色認識はリアルタイムに近い認識速度が必要



探索対象エリア

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

探索対象者認識

探索対象者認識フロー

認識1.探索対象者候補領域の検出



認識2. 探索対象者候補領域に対して、探索対象者か否かの確認



画像より探索対象者の方向を算出し、アプローチ



認識3. 探索対象へアプローチ後、再確認

探索対象者認識の詳細



認識1. 探索対象者候補検出

- 深層学習を用いて探索対象者候補領域を検出
- GoogLeNetで4変数(x,y,width,height)を推測
- 計算時間は0.4secほど



認識2. 候補領域の確認

- 候補領域に対して、探索対象者か否かを確認
- GoogLeNetで探索対象者か否かを分類
- 探索対象を認識した場合、画像より探索対象者へのアプローチ方向を算出する



認識3. アプローチ後の再確認

- 探索対象にアプローチ後、正しくアプローチできたか再確認を行う
- GoogLeNetで探索対象者が写っているか否かを分類

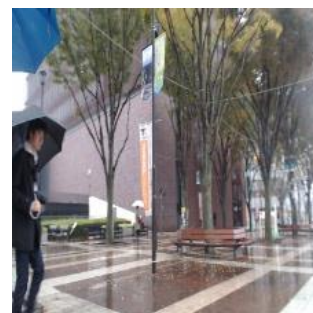
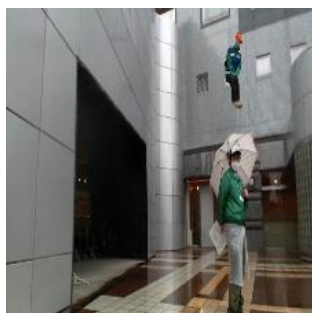
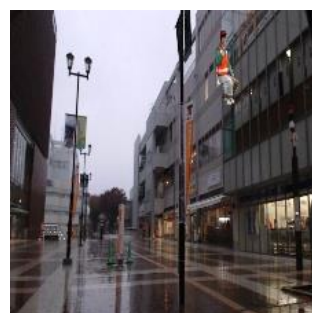
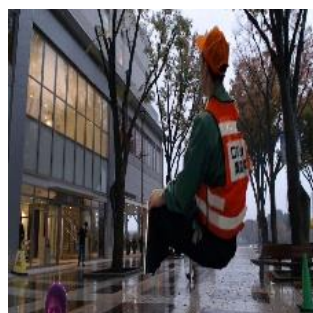
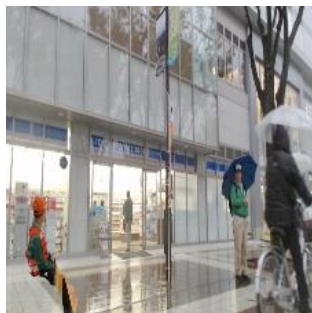
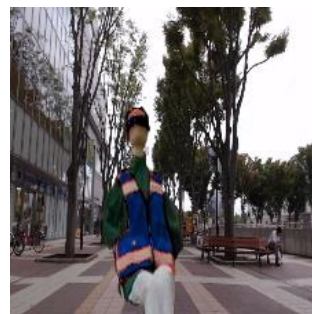
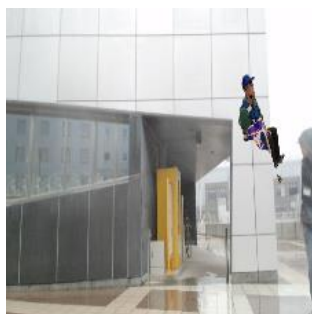
用いた学習データ

- 深層学習では大量の画像が必要である
 - 様々な背景かつ様々な視点の人物など
- 画像処理によって想定される環境を再現
 - 背景画像
 - 探索対象が写っていない画像(9842枚)
 - 手作業で切り出した画像
 - 探索対象画像(40枚)
 - 適当な人の画像(18枚)
 - 安全管理責任者の画像(6枚)
 - 自転車の画像(5枚)



背景画像に画像を合成して学習データを生成

生成した画像例



認識1 探索対象の候補領域の検出

- GoogLeNetに4変数($x, y, width, height$)で学習
 - 画像サイズは 227×227
 - 学習データには8万枚のデータ拡張した画像を使用
 - 探索対象が写っていない画像は $(0, 0, 0, 0)$ を割り当て
- 認識
 - 物体検出ネットワークよりも浅い
 - 計算速度が速い
 - Core i7 6700で0.4sほど
- 使用したフレームワーク
 - NVIDIA社 DISITS
 - Caffe



学習に用いた画像の例(227×227)

認識2 探索対象者検出の確認


- 探索対象の候補領域の検出時に探索対象ではない領域がたまたま検出されていた
 - 候補領域に対して、分類を行うことで精度を高める
- 確認のための学習
 - 探索対象と探索対象以外(背景)の2クラスでGoogLeNetで学習
- 学習データ(4万枚 227x227)
 - データ拡張した画像から探索対象を切り抜いた画像を使用
 - 探索対象画像がはみ出る場合も含める
 - 対策対象以外の画像には背景画像をランダムに切り抜いた画像を使用

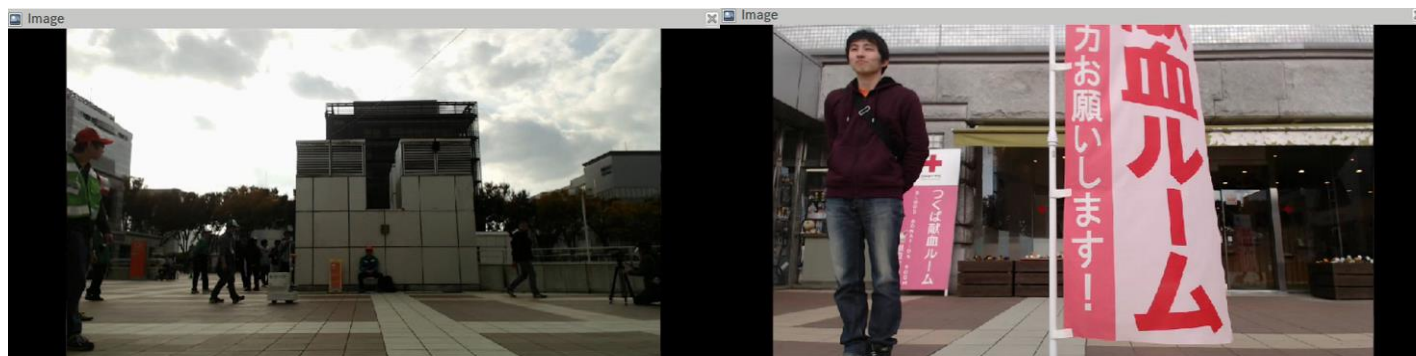


分類の学習に使用した探索対象画像の例

探索対象認識結果

 : 候補領域として切り取られたが判別によって探索対象ではないと認識

 : 探索対象と認識



本走行での認識



本走行後の実験走行

本走行の探索対象者認識結果



- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)

横断歩道走行

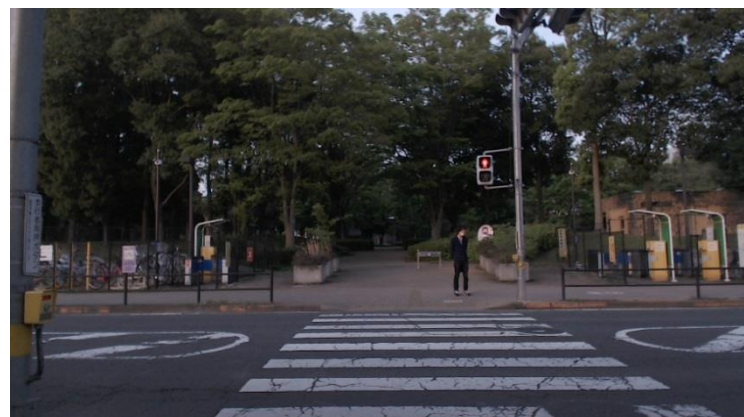
- 押しボタン式の信号機
 - 押しボタンは人が押す
- 20秒以内に渡りきる
 - 緑の時間は15秒、点滅5秒
 - 一般的な信号機より長い



横断歩道周辺の地図

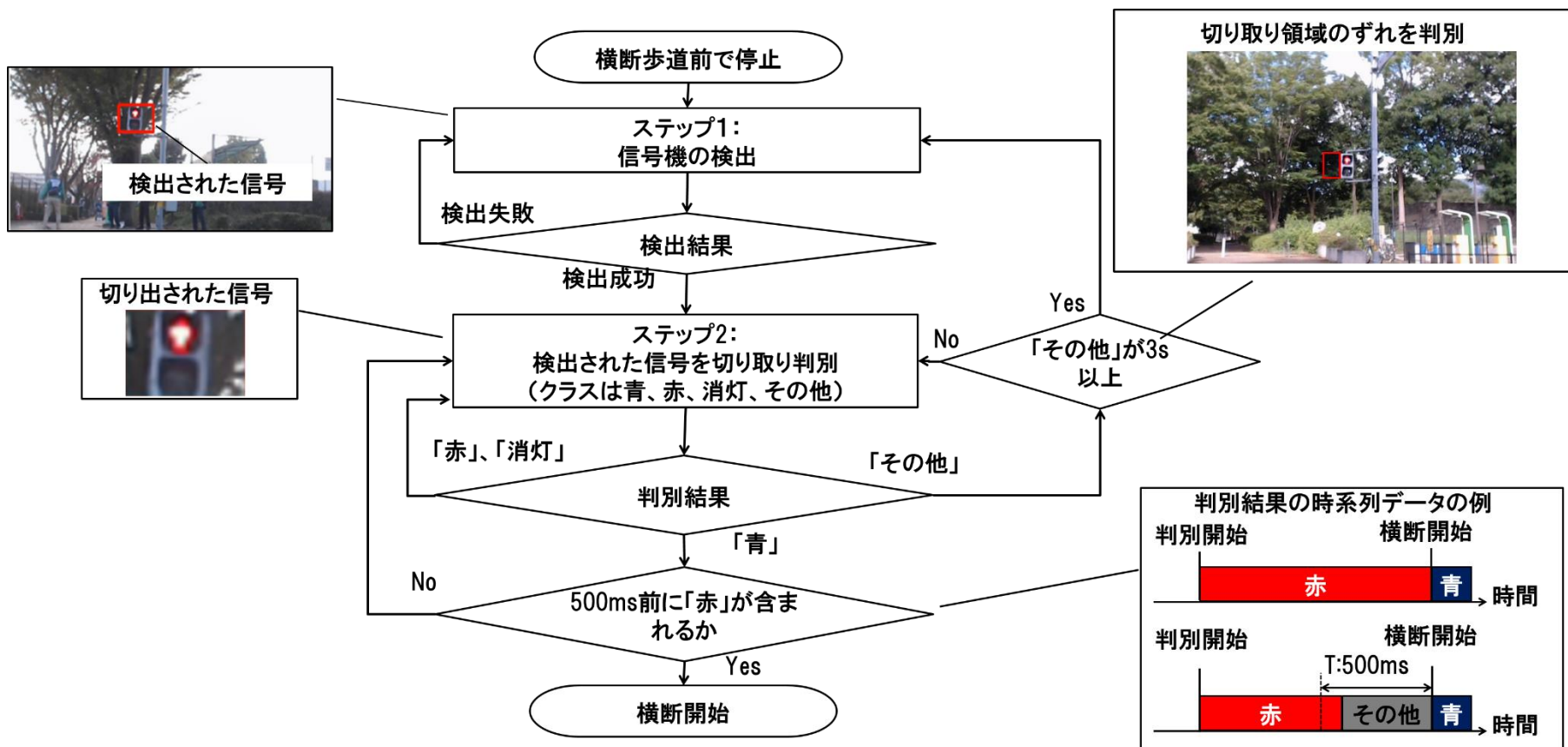


公園側から見た信号機



学校側から見た信号機

信号機認識フロー



ステップ1: 信号機の検出

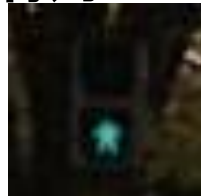
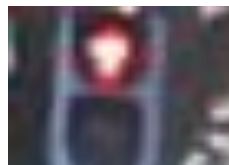
- 信号機の有無、信号を囲む矩形座標出力
- NVIDIA社によって公開されている物体検出ネットワークのDetectNetを使用
 - 1度だけ検出すればよいので計算時間は気にならない
 - 入力画像サイズ864×480で4.4秒かかった
- 学習
 - 複数日に撮影した横断歩道両側から撮影した画像90枚を使用(赤、青半分)
 - 画像サイズは864×480



信号機検出の例

ステップ2: 信号機の判別

- 検出した画像を4クラスに分類
 - 青、赤、消灯、その他(人、自動車などによる隠れ)
- 分類にはGoogLeNetを使用
 - 入力画像サイズを 64×64 に変更(元は 227×227)
→0.1秒で計算可能
- 学習
 - 複数日に撮影した画像を使用
 - 人手で信号機を切り出した
 - 青:7704枚、赤22972枚、消灯:738枚、その他:1401枚



学習データに使用した画像の例

横断歩道を渡る様子



□ : 青信号と認識

□ : 赤信号と認識



- 深層学習を使った4つの実例を紹介した
 - メータ認識、バルブ認識、人物認識、信号機認識
 - 実環境でも高い認識性能
- 自律移動ロボット上のコンピュータでも深層学習で認識が出来る
 - CPU、小型GPU
 - ただし、計算速度を意識する必要がある

- 深層学習の紹介(9分)
 - 深層学習のとは
 - 深層学習の代表的なネットワークとフレームワーク
 - 自律移動ロボットにおける深層学習
- 深層学習を使った実例の紹介
 - Argos Challenge(14分)
 - メータ読み取り
 - バルブ読み取り
 - Tsukuba Challenge(14分)
 - 人物探索
 - 信号機認識
- デモ(7分)