

# 機械学習を用いた路面状況通知システムの設計

2017SE022 入山裕太 2017SE081 鈴木開斗

指導教員:野呂昌満

## 1 はじめに

現在の日本は車両の大型化や交通量の増加などにより道路の老朽化が進んでおり、定期的な点検が必要とされている。高速道路などは点検がされている一方で、県道や市道は十分に点検が行われていない。したがって、道路の利用者自らが路面状況に注意して走行しなければならない事態となっている。

近年、舗装画像からひび割れを自動的に検出する研究が行われている。CNN(Convolutional Neural Network)を用いて路面画像を認識し、ひび割れを検知する研究 [1] や、路面の光尺度から路面の状態を検出する方法の研究 [2] など様々な研究が存在する。これらの先行研究は道路を点検するという点においては比較的高速で正確な検出を可能にしている。しかし、事故を未然に防ぐという点においては即座にひび割れを検出出来ないことから不十分である。

本研究では、機械学習を用いた路面状況を知覚するシステムの設計を通して、物体検出における検出速度向上の方法について考察することを目的としている。

本研究を行うにあたって、技術課題は以下の2つである。

1. CNNを用いた物体検出速度が速い検出手法の提案
2. 提案した手法の妥当性の検証

ひび割れを検出する手法としてCNNを用いて、ひび割れを検出するシステムを設計する。このシステムは、リアルタイム性を考慮する。運転する際にこのシステムを用いることにより、ひび割れが原因の事故等を未然に防ぐことにも繋がる。

## 2 既存研究と背景技術

### 2.1 既存研究

全らは、CNNを用いてアスファルト舗装のひび割れ損傷を評価する手法を開発した [1]。実際の舗装の撮影画像を用いた実験により、ひび割れを高精度に検出でき

ることや、目視による診断と近似した診断ができることを示した。この研究では、GIS(Geographic Information System)上にマッピングするシステムを用いて、損傷状態をわかりやすく可視化することに成功している。これを利用することにより、ひび割れ点検の労力・コストを大きく低下させることに期待がもてる。

MMS(Mobile Mapping System)を取り付けた車両の上部および車両の進行方向左側に搭載されているカメラによる撮影画像を用いている。MMSとは、自動車にデジタルカメラと3次元レーザースキャナを搭載し、道路およびその周辺映像、3次元データを取得することでGISの3次元データの取得を容易に行うことができるシステムである。このカメラは多目的なものであり、舗装路面は画像内の一部にしか写っていないなど、必ずしもひび割れ検出に適した撮影ができていない。全らは、この問題の改善案として車両後部にカメラを設置し、下面を写すことを提案している。これにより、舗装が写る範囲も大きくなる。また、照明条件の調整も可能となる。したがって、細い影が写り、それをひび割れと誤認識するケースを減らすことができ、画像認識の精度も向上する。

### 2.2 背景技術

本研究では、CNNを用いる。また、今後の研究に使うことができるFaster R-CNNについて記述する。以下にそれぞれのNeural Networkについて示す。

#### 2.2.1 CNN

CNNとは多くのデータが局所特徴を持つという性質を利用したNeural Networkである。CNNはデータ中の局所特徴を検出して予測に利用する。また、そもそもどのような局所特徴が存在するのかを訓練データから学習する仕組みを備えている。

#### 2.2.2 Faster R-CNN

Faster R-CNNは特徴マップを抽出する畳み込み層と物体領域を抽出するRPN(Region Proposal Network)に加え、分類、回帰の結果を出力するネットワークで構成されている。

Convolutional Layerで得られた結果をRPNを用い

て、領域を抽出し分類する。Faster R-CNN において RPN が従来の手法から改善された部分である。

従来の手法では、物体候補領域の抽出に時間がかかっていた。その箇所を End-to-end で実施することで従来より高速かつ精度改善を可能にしている。

### 3 路面状況通知システムの設計

路面状況通知システムの全体図を図 1 に示す。

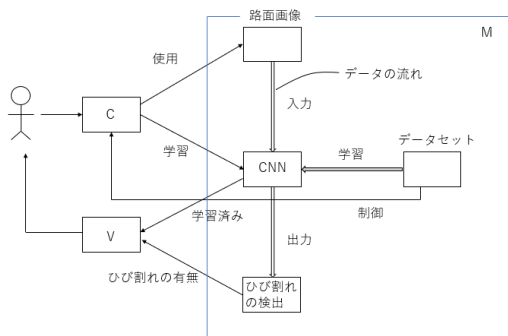


図 1 路面状況通知システムの全体図

図 1 の M は Model, V は View, C は Controller を示す。本研究で扱うシステム全体を MVC システムで設計した。本研究の Model は、事前にデータセットを用いて学習を行った CNN に路面画像を入力する。そして学習済みの CNN を用いて入力した画像にひび割れがあるかひび割れないかの判別を行う。View では、路面画像のデータを入力しひび割れを検出した際の結果をもとにデータが作成され、このデータをこのシステムの利用者に通知する。Controller ではこのシステムの利用者が実際の路面画像を用いて CNN に入力し学習を行うかデータセットを用いて CNN に学習させるかを選択することができる。データセットを用いて学習させるか、このモデルを使用するかを選択はこのシステムの利用者は繰り返す。学習した CNN の結果は View を通してこのシステムの利用者に通知される。本研究では MVC システムの Model にあたる部分を作成する。View, Controller にあたる部分を作成することで車両に搭載して実際に使用することが可能となる。

CNN を用いてリアルタイムで路面状況を検出し通知するシステムを設計する。第 2 章で CNN を用いてひび割れを検出するシステムを記述したが、これは路面の整備を目的としており、本研究の最終目標であるひび割れによる事故を防ぐということとはできない。本研究では、リアルタイムでより速くひび割れを検知することに対する要求に応えることを目的としている。これらの要求を満た

す CNN を用いたシステムを設計する。CNN の畳み込み層を 4 層、プーリング層を 2 層、全結合層を 2 層とする。これらの層の数は、リアルタイム性を実現するために、ひび割れの検出精度が高く検出速度が速くなる値を実験を繰り返すことによって求めた。

このシステムはひび割れのある画像とひび割れない画像を分類しこれらを学習データとして学習させたモデルであり、路面画像を入力とし、ひび割れの有無を出力とする。このシステムに路面画像が入力されてからひび割れの有無を判定するまでの速度を計測し比較することで手法の妥当性を検証する。この妥当性の検証が行われることで、よりひび割れを速く検出することができるので、第 1 章でも述べたようにひび割れが原因の事故を未然に防ぐことができる。

### 4 学習モデルの設計

#### 4.1 学習モデル

CNN を設計するにあたり、画像データの収集を行う。本研究で使用する画像データはインターネット上のものを使用する。インターネット上に公開されているひび割れがある画像とひび割れない画像を分類し 1 つのファイルにまとめ、それを学習データとして学習機に学習させる。ひび割れがある画像 286 枚、ひび割れない画像 412 枚を学習データとして使用する。

CNN を設計するにあたり、画像データの特徴抽出を行う。畳み込み層のフィルタサイズを  $3 \times 3$ 、プーリング層のフィルタサイズを  $2 \times 2$  と設定する。畳み込み層のフィルタサイズは奇数かつ小さいサイズの方が精度が高くなることからこのフィルタサイズとした。プーリング層のフィルタサイズを大きく設定しすぎると、特徴が荒くなるだけでなく、プーリング層のサイズが画像サイズを超えてしまう可能性があることからこのフィルタサイズとした。畳み込み層、プーリング層間の活性化関数は ReLu 関数を用いる。ReLu 関数を用いることにより、関数への入力値が 0 以下の場合には出力値が常に 0、入力値が 0 より上の場合には出力値が入力値と同じ値となるので、精度が向上する。出力層の活性化関数は SoftMax 関数を用いる。これは主に分類問題に用いられる。この関数を用いることにより、出力を確率として学習させることができる。

#### 4.2 パラメータの決定

CNN を設計するにあたり、パラメータの決定を行う。本研究では、パラメータの推定として、損失関数、最適化アルゴリズム、評価関数を以下のように決定した。

- 損失関数：二値交差エントロピー

本研究はひび割れであるか、ひび割れでないかを分類する二値分類モデルである。よって、損失関数に二値分類を行う Neural Network で使われる二値交差エントロピーを用いる。

- 評価関数：accuracy

本研究はひび割れであるか、ひび割れでないかを分類する二値分類モデルである。accuracy は予測結果に対して、予測が正しい割合を表すことから本研究に適していると考え、accuracy を用いた。

- 学習アルゴリズム：adam

adam はパラメータの推定として、一般的に使われる最適化手法であり、勾配の二乗平均と平均を1次モーメントと2次モーメントとして考慮することで、パラメータ毎に適切なスケールで重みが更新される最適化手法である。本研究では他の最適化手法の中でも、学習率の調整問題が最も改善されたこの手法を用いる。

CNN を設計するにあたり、ユニット数、バッチサイズ、エポック数、ドロップアウト率を決定する。ユニット数を100、バッチサイズを64、エポック数を50、ドロップアウト率を0.2とする。ユニット数を50~500、バッチサイズを32~128、エポック数を30~100の範囲で変更し学習を繰り返すことで、表1を最適なパラメータとして決定した。

## 5 実験結果

実験で得られた損失関数と精度のグラフを図2に示す。

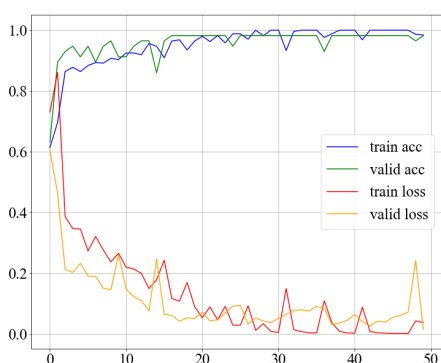


図2 損失関数と精度

図2の青線、緑線はそれぞれ、train用、validation用のデータの精度を意味している。これらの精度はepoch数が増加するにつれて線形的に向上するのではなく、ある一定のepochの時に急激に向上している。本研究の場合、epoch14~17の時に精度が向上している。この現象はディープラーニングで精度が向上する時によくみられる。これは適切な重みを発見した時に生じる。

図2の赤線、黄線はそれぞれ、train用、validation用のデータを解析し出力したものについての損失である。本研究の実験結果では、train用のデータの精度は0.9789、validationのデータの精度は0.9825である。train用のデータの損失は0.1065、validation用のデータの損失は0.0146である。

学習データとは別の路面画像を分類した結果と分類にかかった時間の例を図3に示す。

学習データとは別の路面画像を分類した結果と分類にかかった時間の例を図3に示す。

```
In [66]: runfile('C:/Users/17se081/.spyder-py3/untitled2.py', wdir='C:/Users/17se081/.spyder-py3')
y_pred: [[0.98171799 0.9982821 ]]
number_pred: 1
label_pred: あり
経過時間: 0.23259520530700684

In [67]: runfile('C:/Users/17se081/.spyder-py3/untitled2.py', wdir='C:/Users/17se081/.spyder-py3')
y_pred: [[0.9980135 0.00198649]]
number_pred: 0
label_pred: なし
経過時間: 0.3079257011413574

In [68]: runfile('C:/Users/17se081/.spyder-py3/untitled2.py', wdir='C:/Users/17se081/.spyder-py3')
```

図3 路面画像の分類と経過時間

この実験では、学習モデルの妥当性を検証するために学習データとは別の路面画像を入力した。ひび割れあり、ひび割れなしそれぞれ20枚ずつCNNに入力した。その結果、ひび割れなしに分類されるはずの路面画像が1枚誤検出された。また、画像1枚当たりの平均検出時間はおよそ0.2664秒であった。

## 6 考察

本研究の実験結果と図2より、train用データとvalidation用データの精度はそれぞれ0.9789、0.9825であり、train用のデータの損失は0.1065、validation用のデータの損失は0.0146であった。この結果から、精度が高く、損失が少ないので、概ねCNNを用いてニューラルネットワークにより設計したひび割れありとひび割れ無しの2つの種類に分類するモデルは正確であると考察する。本研究では、ユニット数を50~500、バッチサイズを32~128、エポック数を30~100の範囲で変更し学習を繰り返すことで、グラフのスパイクを防ぐ工夫をした。しかし、エポック数を増加させるとすぐに過学習を起してしまうことから、より精度を向上させ過学習を防ぐために、画像データの量を多量に増やすことや実際に車から撮影されたアングルに近い画像、天候や時間帯など様々な種類の画像を使用することなどが今後の課題としてあげられる。

図3より、学習モデルの妥当性の検証についての考察を述べる。学習に用いた路面画像とは別の路面画像を用

いて分類を行ったが、本来ひび割れのない路面であるが、影をひびと判別しうまく分類できない画像もあった。本研究で扱った影等が写っている画像のようなものが少なかったため、これらの問題が起こってしまったと考察した。様々な学習画像を用意することでより適切な学習ができ、モデルの精度も向上する。路面画像1枚当たりの平均検出時間はおよそ0.2664秒であることから時速50kmの車が3.7m移動する程の時間であるので、検出速度はまだ改良の余地がある。背景技術で記述したFaster R-CNNを用いてモデルを設計することで検出速度が向上する可能性がある。しかし、Faster R-CNNにはカスタム化したRegion Proposalを使用できないという制約が存在することから、ほかの検出手法も調査する必要がある。

Faster R-CNNを用いて設計を行った場合の予想される検出速度は17[fps]である[5]。Fpsとはframe per secondのことである。これを路面画像1枚当たりの平均検出時間に変換すると0.0588秒であり、本研究の実験により得られた0.2664秒よりおよそ4.5倍検出速度が速くなることが予想される。よって、Faster R-CNNを用いることによってより検出速度が向上すると考察する。本研究の背景技術には記述していないが、Faster R-CNN以外にも検出速度が速い検出手法は存在する。この検出手法を用いることで、より検出速度や検出精度が向上する可能性があるため、今後の研究で使用することを検討する必要がある。

本研究では図1のMVCシステムのModelにあたる部分を作成した。このモデルはデータセットを用いてCNNに学習させ、路面画像を入力しひび割れの有無を出力とした。今後の研究で、View、Controllerにあたる部分を作成することで車両に搭載して実際に使用することが可能となる。運転するとき自動録画を行い、ひび割れの情報を周囲に送信することを考えると図1のシステムでは不十分である。位置情報を取得することでこの問題が解決され、より実用性の高いシステムを作成できると考察する。

## 7 おわりに

本研究では、機械学習を用いてリアルタイム性が高く正確な検出を行うことを課題とした。先行研究では路面の点検を目的として機械学習を用いていた。このことから先行研究ではリアルタイム性より精度の高さを重要視していた。

我々は、機械学習を用いた路面状況を通知するシステムの設計を通して、物体検出における検出速度向上の方法について考察することを目的とした。物体検出における検

出速度が向上することで、従来の路面の点検に利用するだけでなく、事故を未然に防ぐことができる。

CNNを用いて路面画像からひび割れを検出するモデルを設計し、実験を行った。このモデルは、路面画像からひび割れの有無で2種類に分類するモデルである。

実験の結果、本研究で設計した学習モデルは精度がおよそ98%と十分であった。学習モデルの妥当性を検証した結果、95%が正しく分類された。また、画像1枚当たりの平均検出時間はおよそ0.2664秒と高速な検出を可能にした。

今後の課題として、より正確かつ高速検出可能なモデルを設計する。このようなモデルを設計するために、より多くのデータを集めることやFaster R-CNNなどの高速検出可能な手法を用いたモデルを設計し、本研究で設計したモデルと比較していくことが重要である。また、ひび割れありの分類を細分化することで、より危険なひび割れのみを検出することができ、実装時の実用性向上に繋がると考察する。

## 参考文献

- [1] 全邦釘, 井後敦史, 南免羅裕治, 黒木航汰, 大窪和明, “車載カメラにより撮影された舗装画像からのディープラーニングによるひび割れ率評価”, 土木学会論文集E1(舗装工学), Vol. 73, No. 3 (舗装工学論文集第22巻), 2017.
- [2] 久野徹也, 杉浦博明, 吉田潤一, “車載カメラによる路面状態検出方法の検討”, 電子情報通信学会論文集, Vol. J81-D-2, No. 10, pp. 2301-2310, 1998.
- [3] 速水 悟, “畳み込みネットワークを押さえる”, <https://xtech.nikkei.com/atcl/learning/lecture/19/00065/00003/>, 2019.
- [4] Shaoqing Ren, Kaiming He, Ross Grishick, and Jian Sun, “Faster R-CNN: Towards Real-Time Object Detection With Region Proposal Networks”, arXiv:1506.01497v3 [cs.CV] 6 Jan 2016.
- [5] Xiangteng, Yuxin Peng, Junjie Zhao, “Fine-grained Discriminative Localization via Saliency-guided Faster R-CNN”, arXiv:1709.08295v1 [cs.CV] 25 Sep 2017.