



DL4US 最終課題 発表

— CNNを用いたStreet分類とその深層特徴分析

門脇 宗平 (かどわき しゅうへい)

京都大学総合人間学部

github: aviatesk

TOC

1. 概要
 - 1.1. やったこと
 - 1.2. 発表したいこと
2. 実験設計
 - 2.1. 使用データ
 - 2.2. 使用モデル
3. 結果・考察
 - 3.1. 転移学習とFine-tuning
 - 3.2. デモ・深層特徴解析
4. まとめ

TOC

1. 概要

1.1. やったこと

1.2. 発表したいこと

2. 実験設計

2.1. 使用データ

2.2. 使用モデル

3. 結果・考察

3.1. 転移学習とFine-tuning

3.2. デモ・深層特徴解析

4. まとめ

1.1. やったことーモチベーション



バンクーバーっぽい

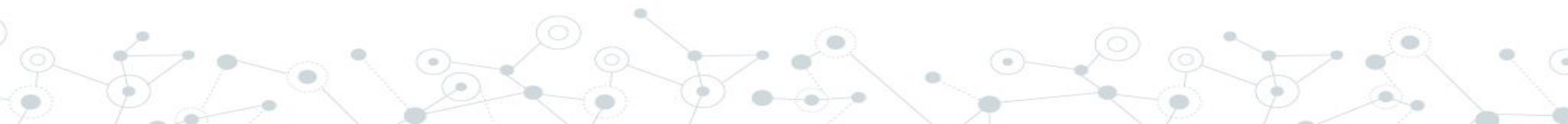
1.1. やったことーモチベーション



東京っぽい

1.1. やったこと

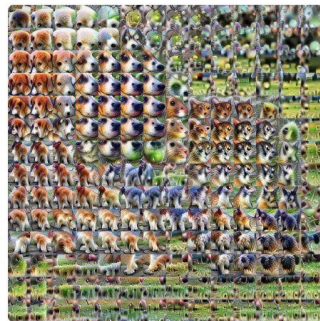
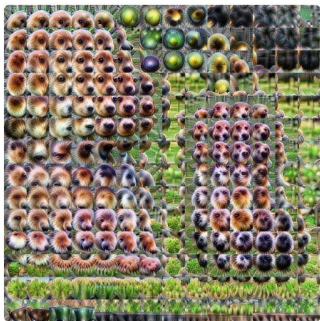
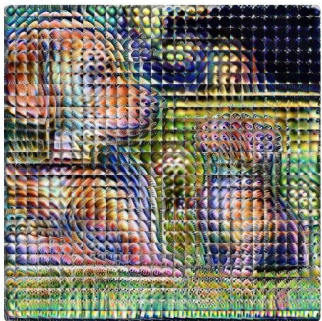
なにをもって、
「〇〇の道っぽい」
と感ずるのか知りたい！！



1.1. やったこと ー 深層特徴分析

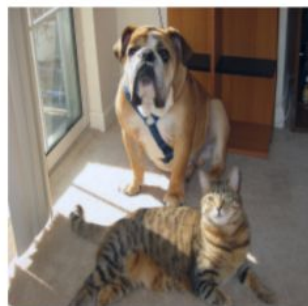
CNNのConv layerには面白い情報がたくさん含まれている(深層特徴, deep feature)

- Feature visualization
- Attribution analysis



[The Building Blocks of Interpretability \(http://places2.csail.mit.edu/index.html\)](http://places2.csail.mit.edu/index.html)

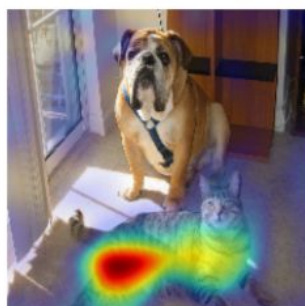
1.1. やったこと — GradCAM



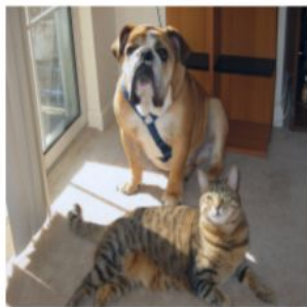
(a) Original Image



(b) Guided Backprop 'Cat'



(c) Grad-CAM 'Cat'



(g) Original Image



(h) Guided Backprop 'Dog'



(i) Grad-CAM 'Dog'

Grad-CAM

- 分かりやすい
 - 情報量が少ない
- 実装も簡単

[Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization](https://arxiv.org/abs/1610.02391)
(<https://arxiv.org/abs/1610.02391>)

1.1. やったこと

各都市のStreetの画像を収集・選定



CNNで各都市のStreetの分類を学
習



獲得した深層特徴を可視化

1.2. 発表したいこと

- 転移学習とFine-tuning
 - 転移元ネットワークの学習ドメイン
 - 転移学習 vs. Fine-tuning
 - 転移学習の有効性
- 深層特徴の解釈について
 - デモ
 - 問題点・ほかのアプローチ

TOC

1. 概要

1.1. やったこと

1.2. 発表したいこと

2. 実験設計

2.1. 使用データ

2.2. 使用モデル

3. 結果・考察

3.1. 転移学習とFine-tuning

3.2. デモ・深層特徴解析

4. まとめ

2.1. 使用データ

今回扱いたいような”Street”の画像が国や街ごとにラベル付けされているデータセットが見つけれなかった



[Google Image Search](#)の検索結果からStreet画像と街を紐づけする

- e.g.) New York street photo -art -fashion -food
- [icrawler](#): 簡単にクロールできるライブラリ

2.1. 使用データ

合計10都市についてそれぞれ2000枚ほどのStreet画像をクロージング

- 西洋) ロンドン、パリ、モスクワ、バンクーバー、ニューヨーク
- 東洋) 北京、シンガポール、ソウル、京都、東京
- 今回のStreetの概念に適するもののみを(目視で)選定
 - 間違いなく今回一番つらかった部分
- 余白をcrop, 重複などを削除

結果各都市400ほどのサンプル、合計サンプルサイズ4000のデータセットを作成した

- 多分なバイアス
- 多分な誤り

TOC

1. 概要
 - 1.1. やったこと
 - 1.2. 発表したいこと
2. 実験設計
 - 2.1. 使用データ
 - 2.2. 使用モデル**
3. 結果・考察
 - 3.1. 転移学習とFine-tuning
 - 3.2. デモ・深層特徴解析
4. まとめ

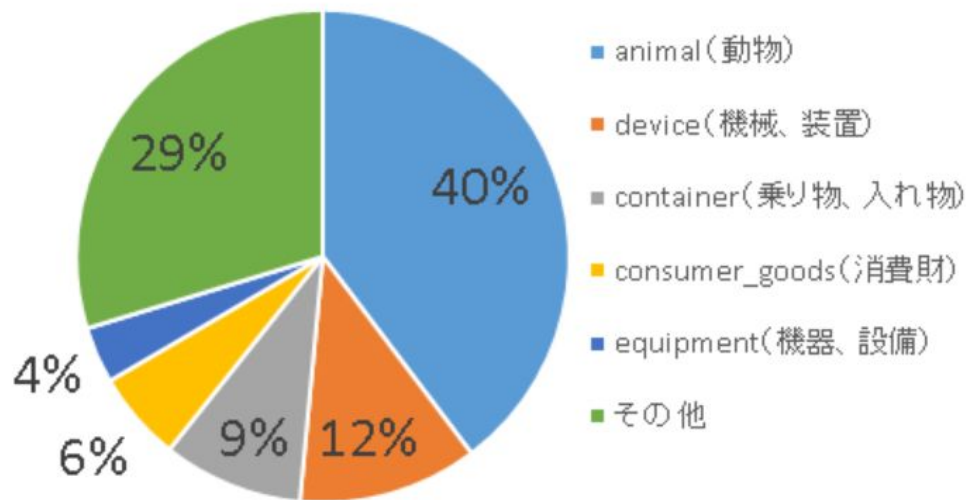
2.2. 使用モデル

ImageNetの中身

今回やりたいこととは学習されたドメインがかなり違う

[ImageNet\(ILSVRC2012\)データセット](http://places2.csail.mit.edu/index.html)
(<http://places2.csail.mit.edu/index.html>)

分類(WordNet第7階層)の構成比率



分類名	クラス例	件数
animal(動物)	ベルシャ猫、ワシなど	397
device(機械、装置)	扇風機、車輪など	118
container(乗り物、入れ物)	客車、花瓶など	92
consumer_goods(消費財)	着物、靴下など	59
equipment(機器、設備)	コピー機、携帯電話など	37
その他	-	297

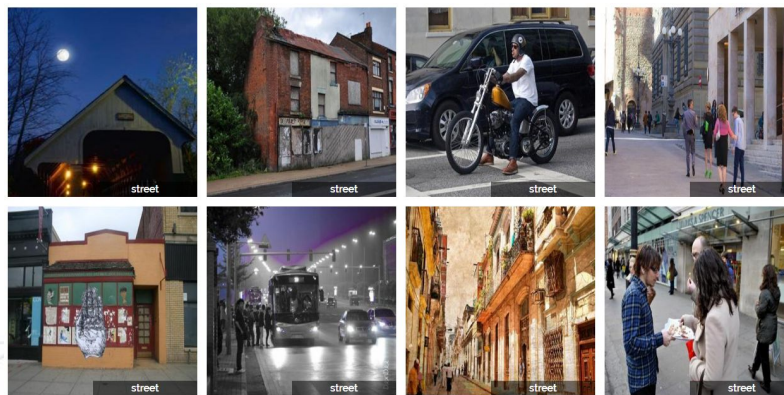
2.2. 使用モデル

Places dataset

- 400以上の**Scene-Category**を含む1000万枚以上の画像
 - Streetなども含まれる
- 今回やりたいことと近そう
- VGG16アーキテクチャで学習させたCNNも公開



[Places: A 10 million Image Database for Scene Recognition](http://places2.csail.mit.edu/index.html)
(<http://places2.csail.mit.edu/index.html>)



2.2. 使用モデル

- ベースライン
 - VGG16-Places365の最後のGlobal Average Pooling Layerの出力にSVMを掛けたもの
 - ランダムな初期値から学習させたVGG16アーキテクチャのCNN
- 転移学習モデル
 - VGG16-Places365の転移学習モデル
- Fine-tuningモデル
 - VGG16-Places365のFine-tuning
 - VGG16-ImageNetのFine-tuning

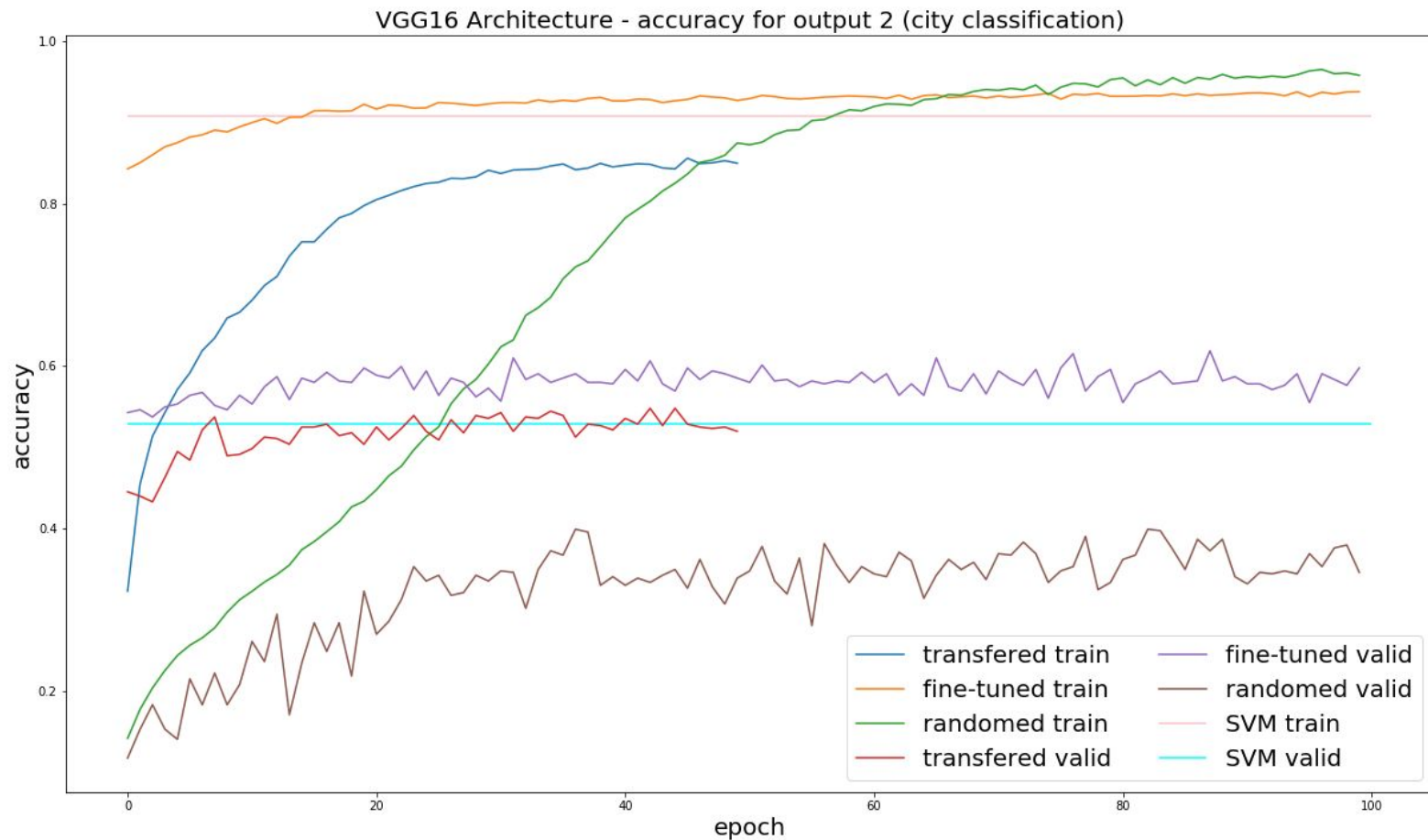
補足

- 転移学習(Transfer-learning)
 - 転移元ネットワークの(最後のConv blockなどの)出力を用いる
 - 転移元ネットワークの重みは再学習しない(freezed)
- Fine-tuning
 - 転移元ネットワークの(後ろの方のConv blockに関してだけ)重みも再学習させる
 - 最適化には小さい学習係数でSGDなどを用いる
- 今回の最終出力は2種類
 - どの街か(10クラス分類)
 - 東洋か西洋か(2クラス分類)

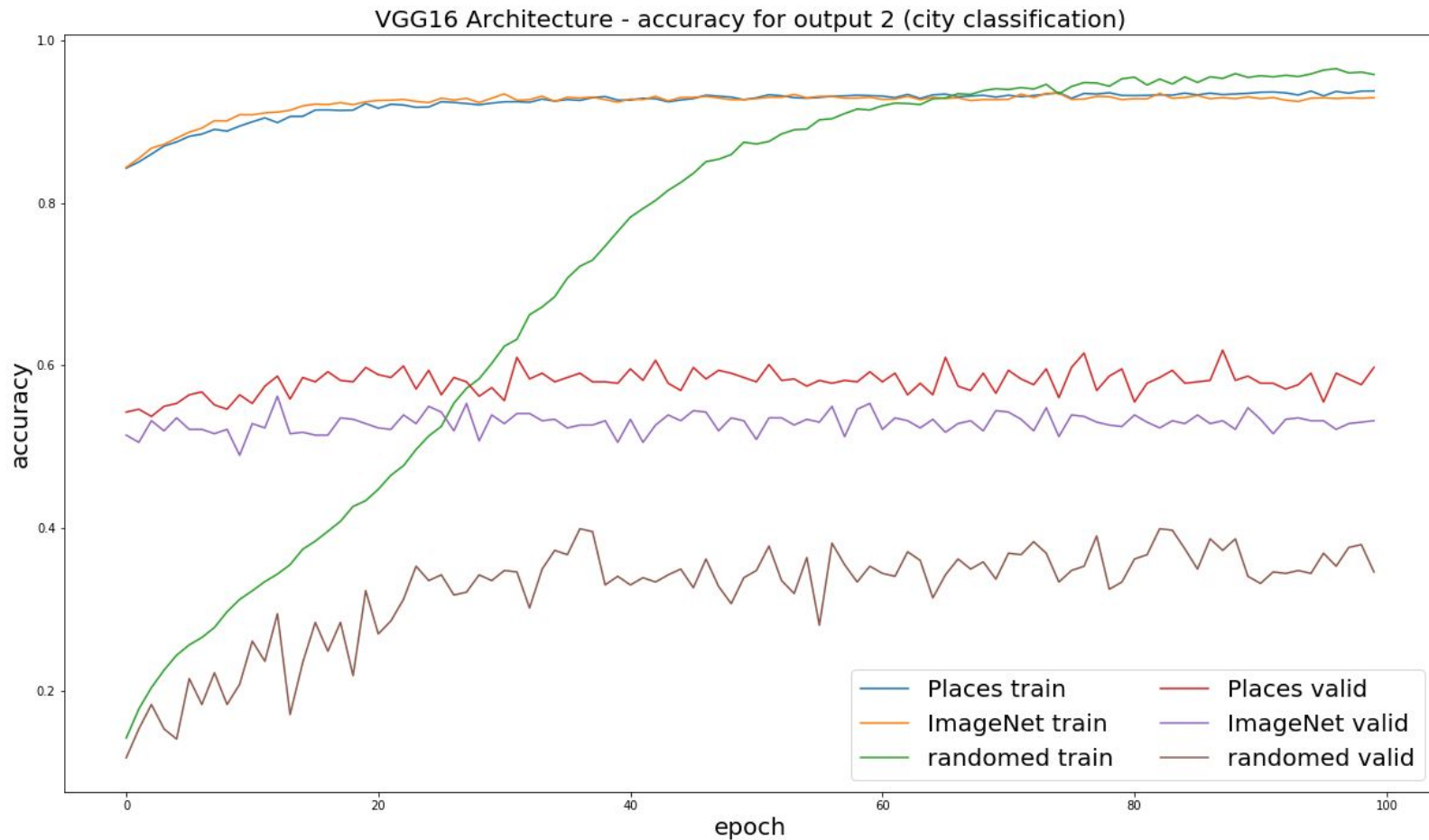
TOC

1. 概要
 - 1.1. やったこと
 - 1.2. 発表したいこと
2. 実験設計
 - 2.1. 使用データ
 - 2.2. 使用モデル
3. **結果・考察**
 - 3.1. **転移学習とFine-tuning**
 - 3.2. デモ・深層特徴解析
4. まとめ

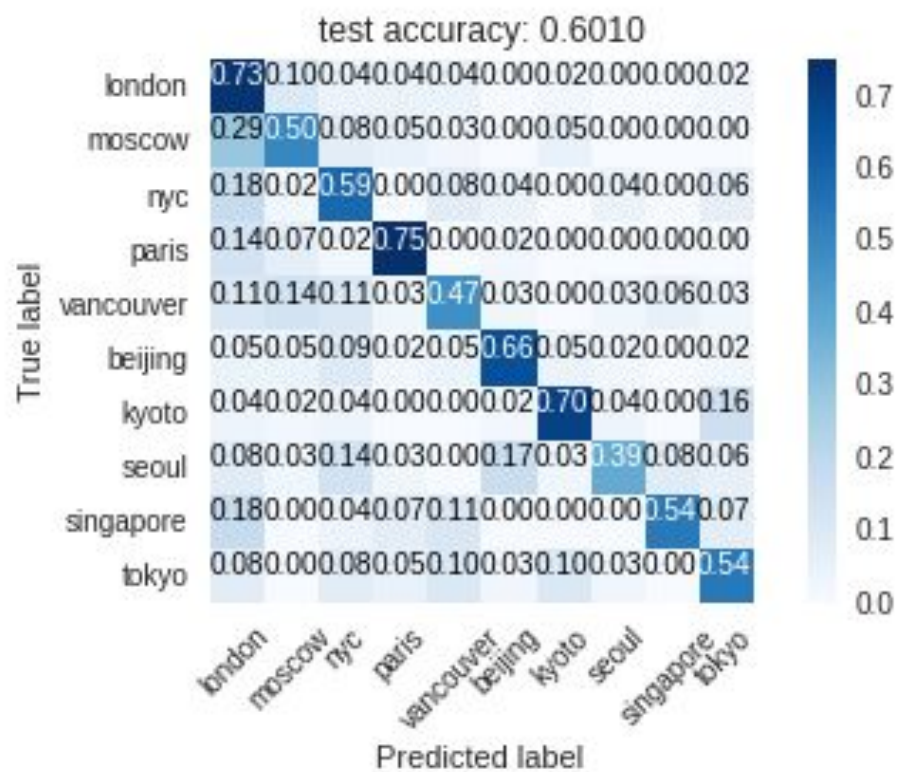
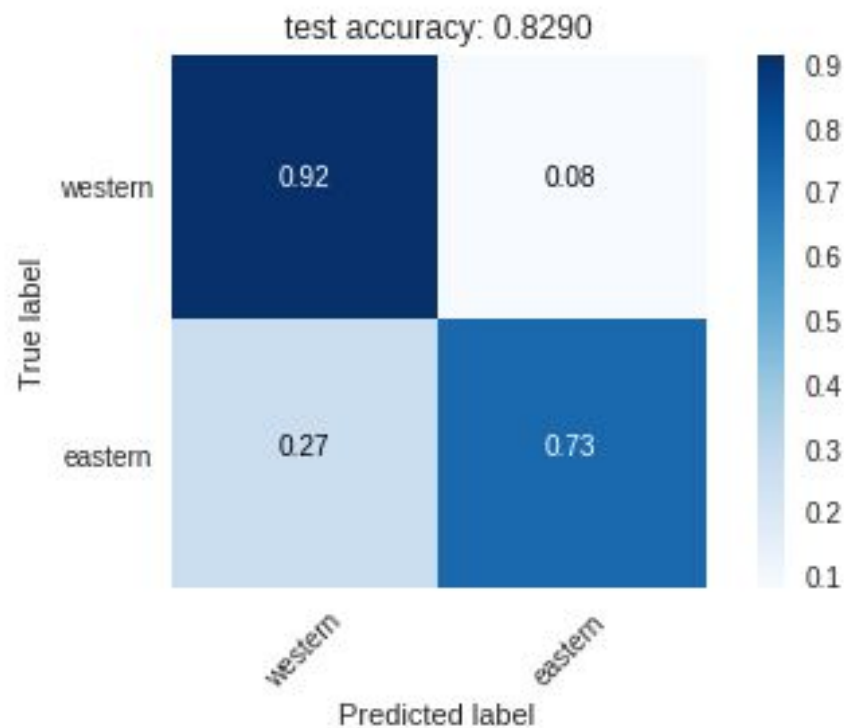
3.1. 結果 — 転移学習 vs. Fine-tuning



3.1. 結果 — Places vs. ImageNet



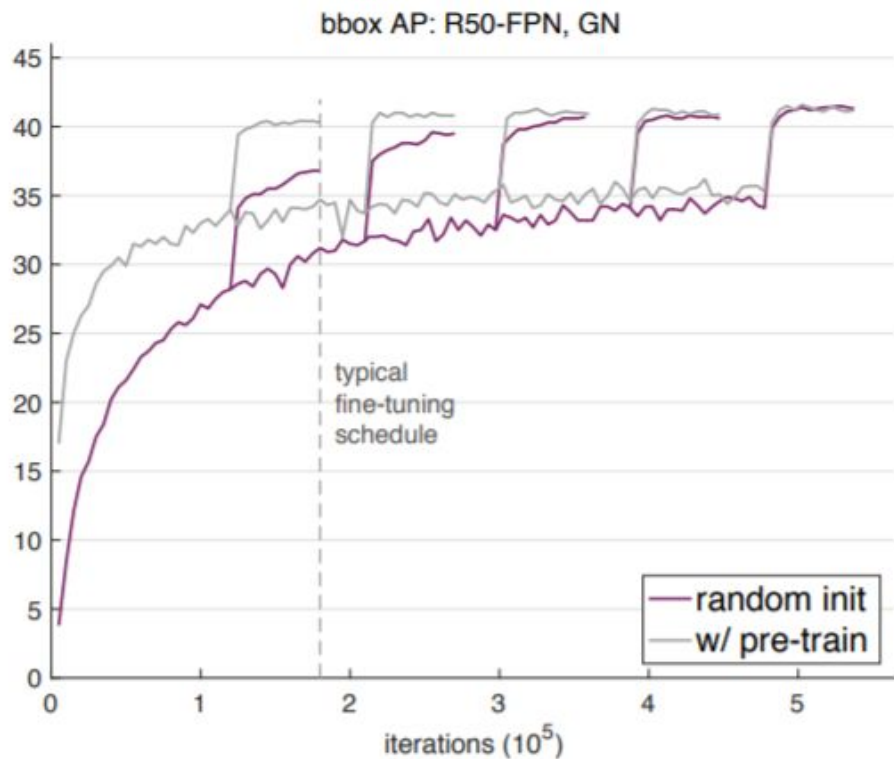
3.1. 結果



3.1. 考察

- 今回の規模のデータセットに対しては転移学習およびFine-tuningが有効だった
 - 転移学習 < Fine-tuning
 - ランダムな重みから学習させると過学習してしまう
- 仮説通り、ImageNet-CNNよりもPlaces-CNNを用いた方が結果が良かった
 - 学習ドメインの共通性

3.1. 考察 ……一方で



Rethinking ImageNet Pre-training (<https://arxiv.org/pdf/1811.08883.pdf>)

- 十分な学習時間を取れば、重みをランダムに与えたネットワークでも転移学習 / Fine-tuningと同じ程度の汎化能力・精度を獲得できる
 - ランダム初期値からの学習時間 \approx ImageNetの事前学習時間 + Fine-tuning分の学習時間
- 学習するデータが小さくないときに限る
 - 論文はCOCOの10% ($\sim 20k$ images)
- ImageNet Pre-trained CNNの学習タスク (classification)と、目的の学習タスク (e.g. 論文ならobject segmentation)が違う場合は、事前学習の利点は低くなる

TOC

1. 概要
 - 1.1. やったこと
 - 1.2. 発表したいこと
2. 実験設計
 - 2.1. 使用データ
 - 2.2. 使用モデル
3. 結果・考察
 - 3.1. 転移学習とFine-tuning
 - 3.2. デモ・深層特徴解析**
4. まとめ

3.2. 深層特徵可視化

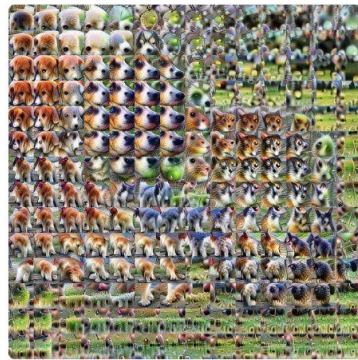
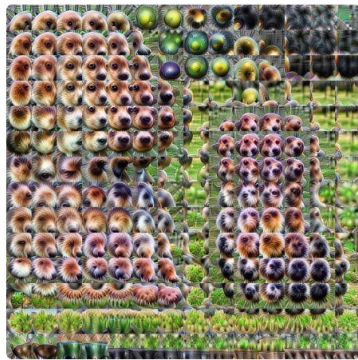
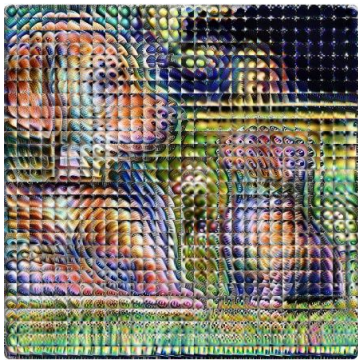
(demo)

3.2. 問題点

- Sample-wiseな可視化しかできないので、データセット全体に渡る深層特徴を可視化できない
 - “..., because they interpret their results back onto the input image, they miss the opportunity to communicate in terms of the rich behavior of a network’s hidden layers.”
[\(The Building Blocks of Interpretability \(https://distill.pub/2018/building-blocks/\)\)](https://distill.pub/2018/building-blocks/)
- Attributionだけで獲得した深層特徴って可視化できるのか？そもそもAttribution methodは信頼できるのか？
[THE \(UN\)RELIABILITY OF SALIENCY METHODS \(https://arxiv.org/pdf/1711.00867.pdf\)](https://arxiv.org/pdf/1711.00867.pdf)

3.2. Future works

The Building Blocks of Interpretability
(<https://distill.pub/2018/building-blocks/>)



3.2. Future works

The Building Blocks of Interpretability

(<https://distill.pub/2018/building-blocks/>)

- feature visualization, attribution analysis, matrix factorizationなどを組み合わせた、深層特徴を可視化するゴージャスなインターフェース
- 情報が多すぎる
- CNNの獲得したfeatureの解釈が難しい
 - 言語化できない、すると意味が失われる


4. まとめ

- 画像を収集、データセット作成
 - Google Image Search, icrawler
 - 大変、バイアスすごい
- 小規模データセット: Fine-tuning
- 転移元のネットワークの学習ドメイン、学習タスクに注意
- 深層特徴の解釈: 面白い、けど難しい
 - 情報を保ちつつ、いかにHuman-scaleに落とし込むか
 - Grad-CAMは簡単だが、情報損失が大きい


References

- [The Building Blocks of Interpretability \(http://places2.csail.mit.edu/index.html\)](http://places2.csail.mit.edu/index.html)
- [Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization \(https://arxiv.org/abs/1610.02391\)](https://arxiv.org/abs/1610.02391)
- [Google Image Search](#)
- [icrawler](#)
- [ImageNet\(ILSVRC2012\) データセット \(http://places2.csail.mit.edu/index.html\)](http://places2.csail.mit.edu/index.html)
- [Places: A 10 million Image Database for Scene Recognition \(http://places2.csail.mit.edu/index.html\)](http://places2.csail.mit.edu/index.html)
- [Keras | VGG16 Places365 - VGG16 CNN models pre-trained on Places365-Standard for scene classification \(https://github.com/GKalliatakis/Keras-VGG16-places365\)](https://github.com/GKalliatakis/Keras-VGG16-places365)
- [Rethinking ImageNet Pre-training \(https://arxiv.org/pdf/1811.08883.pdf\)](https://arxiv.org/pdf/1811.08883.pdf)
- [Grad-CAM implementation in Keras \(https://github.com/jacobgil/keras-grad-cam\)](https://github.com/jacobgil/keras-grad-cam)
- [THE \(UN\)RELIABILITY OF SALIENCY METHODS \(https://arxiv.org/pdf/1711.00867.pdf\)](https://arxiv.org/pdf/1711.00867.pdf)



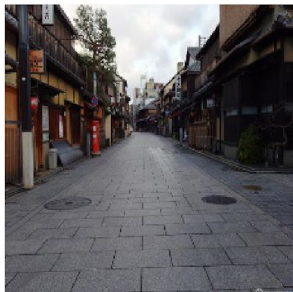
A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. Some nodes are highlighted with blue circles, and others with blue dots.

おわり

A decorative network diagram in the bottom-right corner, featuring a complex web of interconnected nodes and lines. Some nodes are highlighted with blue circles, and others with blue dots.

Appendix

Kyoto



Kyoto



Kyoto



Kyoto



Kyoto



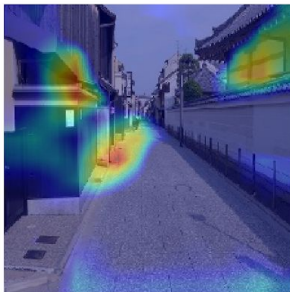
'Kyoto': 98.60



'Tokyo': 61.30



'Kyoto': 99.83



'Kyoto': 98.99



'Tokyo': 87.74



Appendix

NYC



NYC



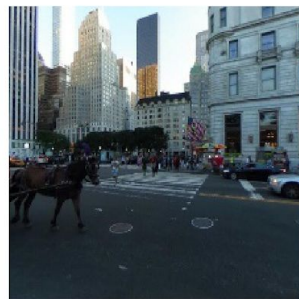
NYC



NYC



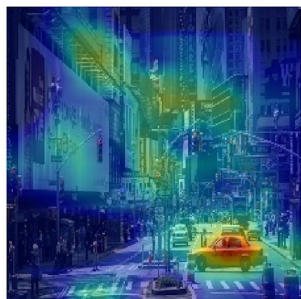
NYC



'London': 96.34



'NYC': 100.00



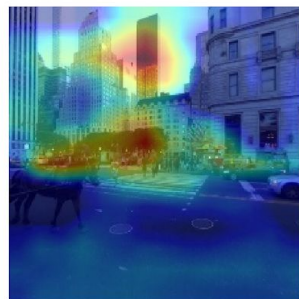
'NYC': 99.99



'Tokyo': 47.84

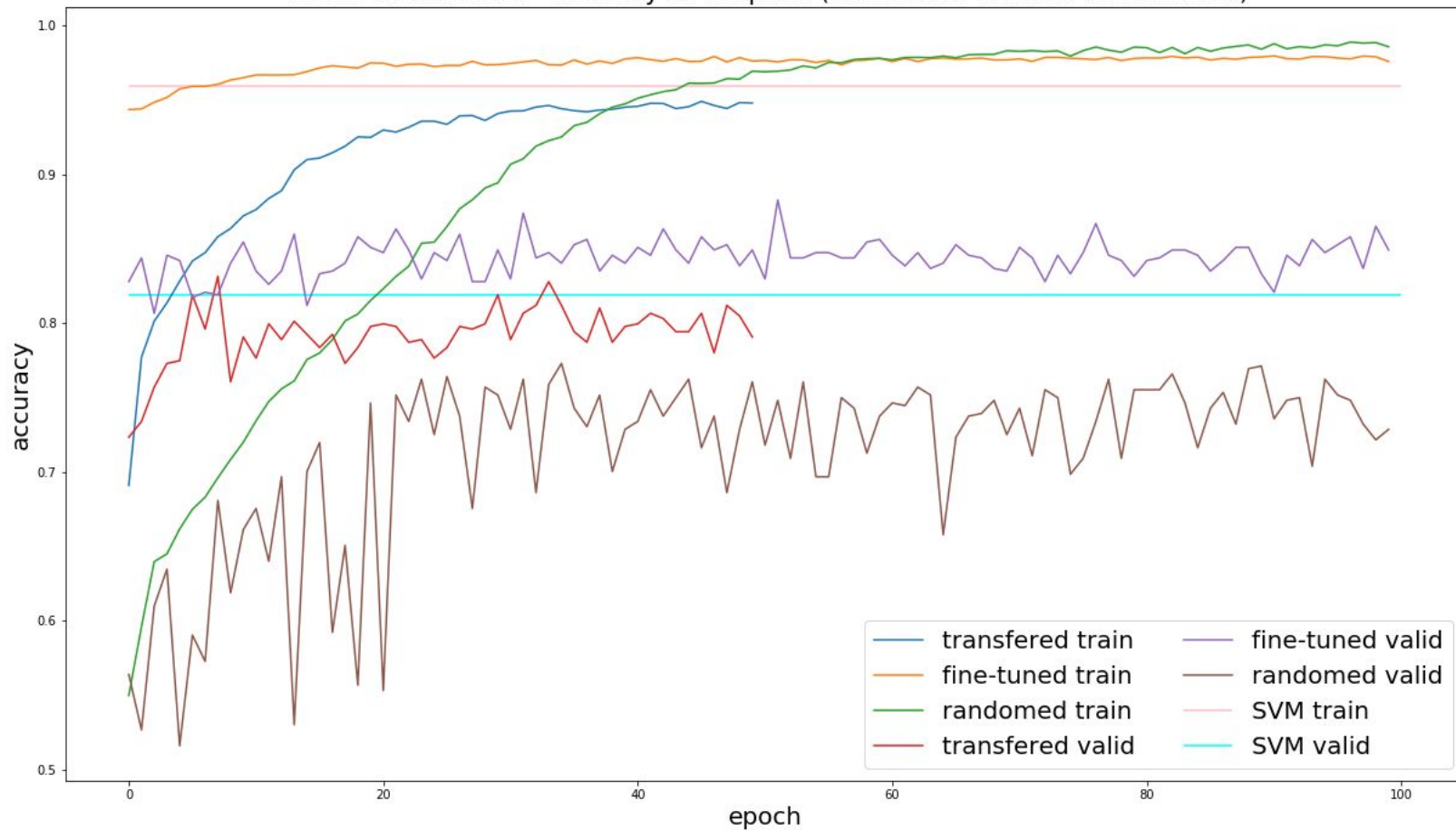


'Seoul': 81.06



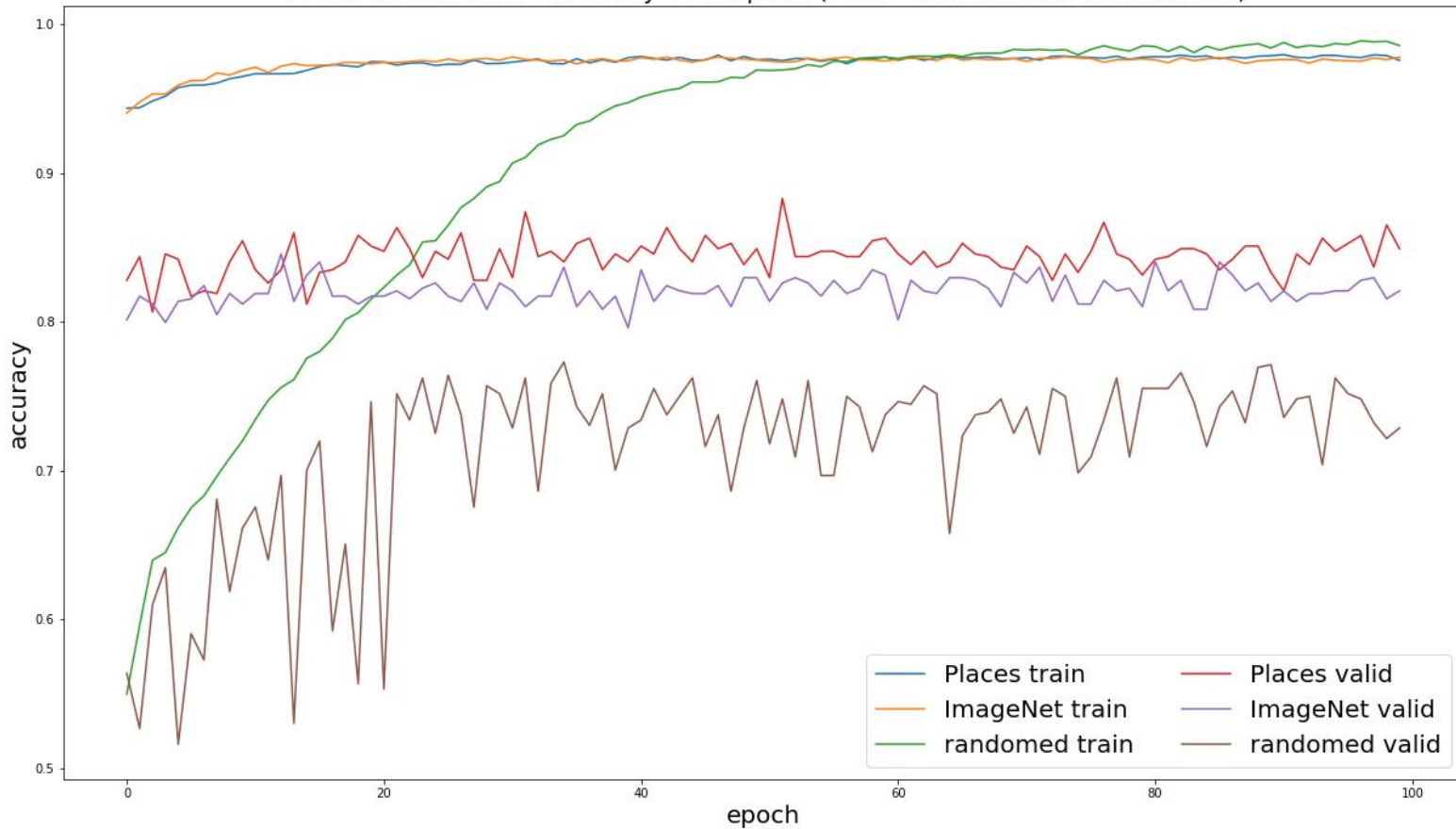
Appendix

VGG16 Architecture - accuracy for output 1 (Eastern vs. Western classification)



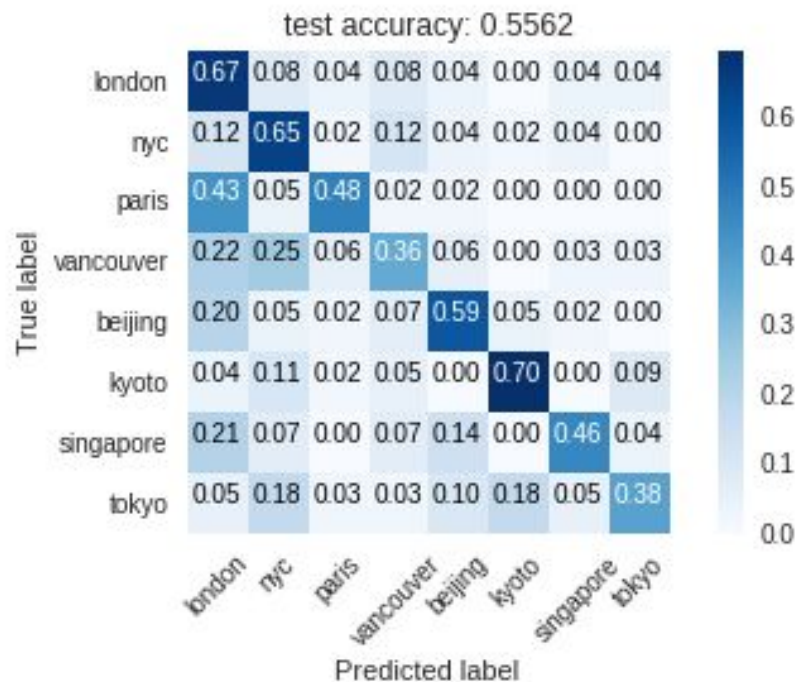
Appendix

VGG16 Architecture - accuracy for output 1 (Eastern vs. Western classification)



Appendix

8 cities classification



10 cities classification

