



1. 研究概要

アプリの特徴

- 全処理アプリ内完結 (サーバ不要)
マルチスレッド, 高速フレームワーク利用による高速化
マルチスケールによる任意画像サイズ対応
必要メモリの大幅な削減(省メモリ化)
各種IT機器への組込容易

例: 101種類認識

- 認識時間は30ms台の高速認識
上位5位以内で93.5%の高精度認識



2. 提案内容

誰でも作れる高速・高精度な物体認識&変換iOSアプリ

~アプリ作成の一連の流れ~

我々の研究室で開発

学習データを用意

Caffe(Chainer)で学習

Caffe2C(Chainer2C)でC言語コードを自動生成

アプリのGUI作成

高速認識(変換)エンジンに組込

完成!

- Caffe2C / Chainer2C
- Caffe(Chainer)で学習したパラメータからモバイルで実行可能なC言語コードを自動生成
高速認識(変換)エンジン
- マルチスレッド, 高速フレームワーク利用による高速化
- マルチスケールCNNによる任意サイズの入力画像に対応
- 画像サイズ調整による, 認識精度と実行速度のバランス調整

学習データさえあれば, どんな認識アプリでも作成可能!

3. DeepXCam (認識版: 2000, food, bird, etc.)

DCNNの学習

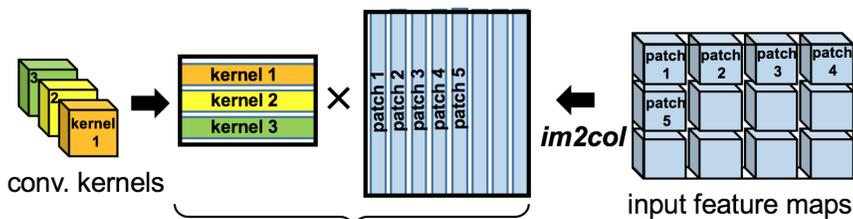
- モバイルの実装を考慮して, Network-In-Network(NIN)[1]を使用
- パラメータ数とメモリ削減

Table with 4 columns: Network In Network [1], Parameters, Memory, Classification Rate (Top-5). Rows include AlexNet, NIN(4L+BN), and NIN(5L+BN).

- 2000種類(ImageNet+Foodカテゴリ)プレトレーニング
UEC-Food(約1万枚)+非食事1万枚でファインチューニング

畳み層の高速化 => GEMMの高速化

- 画像の3次元配列を2次元配列に変換(im2col)
- BLAS(iOS: Accelerate Framework, Android: OpenBLAS)
- NEON命令(同時に4つの32bit単精度浮動小数点のSIMD演算命令)
- iOS: 2Core*4=8演算, Android: 4Core*4=16演算同時実行



GEMM: generic matrix multiplication (=conv. layer computation)

5. 認識性能と認識時間の比較

食事101種類認識性能 (5-fold CV)

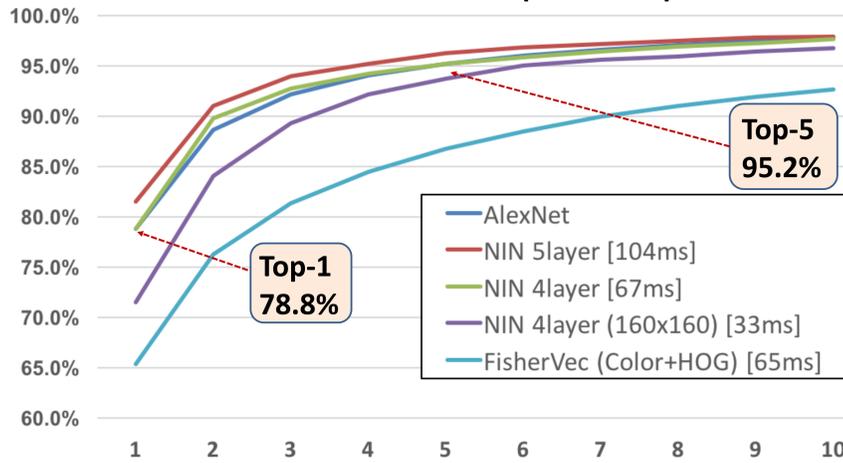


Table comparing recognition performance (Top-1, Top-5, recognition time) for food 101, general objects 2000, birds 200, flowers 17, dogs 100, and omurice 5.

食事101種類と一般物体2000種類とで認識時間にほとんど差は無い
認識カテゴリ数を増やしても, 処理時間はそのまま!

6. iOSとAndroidの特性分析

iOSとAndroidの認識時間の差異

Table comparing recognition times for iOS (NEON, BLAS) and Android (BLAS, OpenBLAS) on different devices.

認識時間と認識精度のTrade-off

Table showing the trade-off between recognition time and accuracy (top-5) for different device resolutions and models.

iOS: NEON < BLAS

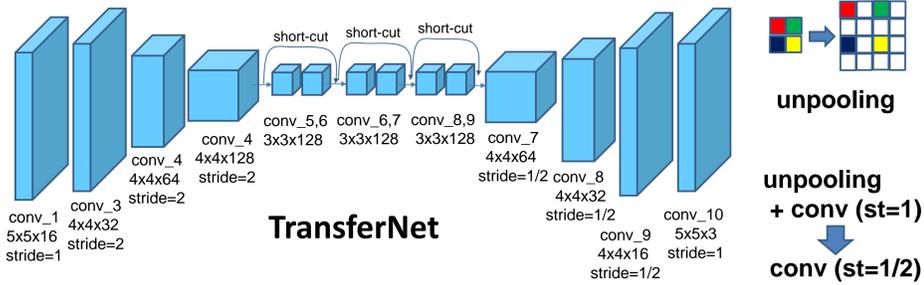
Android: BLAS < NEON

iOS/Androidで最適な高速化手法は異なる

4. DeepStyleCam (画像スタイル変換)

Fast Neural Style Transfer [2] をマルチスタイルに拡張.

- Neural style transferがfeed-forwardのみで高速で可能.
TransferNet (画像変換ネットワーク)を,
- 入力画像と出力画像のVGG19 conv3_3 activationが同一
- 入力画像のgram matrixが style imageのgram matrixと同一となるように学習.
1つのTransferNetが複数スタイルを学習. 重ねあわせ可能.
Deconvolutionは, Unpooling + convolution で実装.



250x250の場合, 乗算回数1,303,800,800回(13億回)
パラメータ数1,250,835個(125万個)
180ms (iPad Pro)
200ms (iPhone SE)

食事認識システム公開中!

現在, iOS/Android/Twitterの3種類

食事認識iOSアプリ
- ダウンロード・デモ
http://foodcam.jp/
https://www.youtube.com/watch?v=em2xzqxSYEA

食事認識Androidアプリ
- ダウンロード・デモ
http://foodcam.jp/

食事認識bot
@fooding_bot宛に食事画像URLをtweet replyで認識結果が送られてきます

[1] M. Lin, and Q. Chen, and S. Yan. Network In Network. Proc. of International Conference on Learning Representations, 2014.

[2] J. Johnson, A. Alahi, L. Fei-Fei: Perceptual Losses for Real-Time Style Transfer and Super-Resolution, arXiv:1603.08155, 2016.



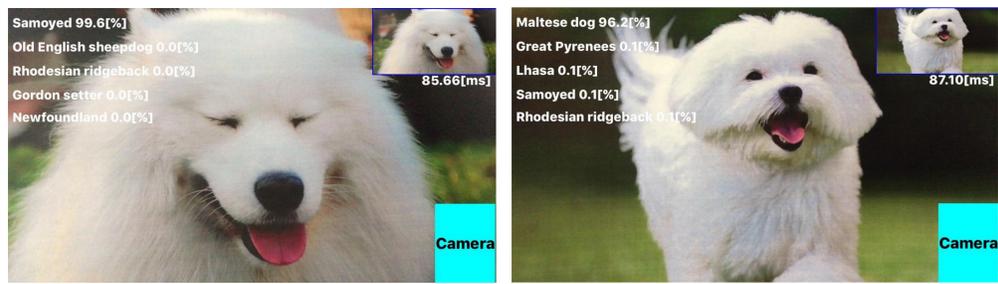
1. DeepFoodCam



- ・101種類食事認識アプリ(食事100種類+非食事)
- ・CNN学習: 食事画像1万枚(UECFood-100), 非食事画像1万枚(Twitter)

認識精度	Top-1	Top-5
食事101種類	78.8%	95.2%

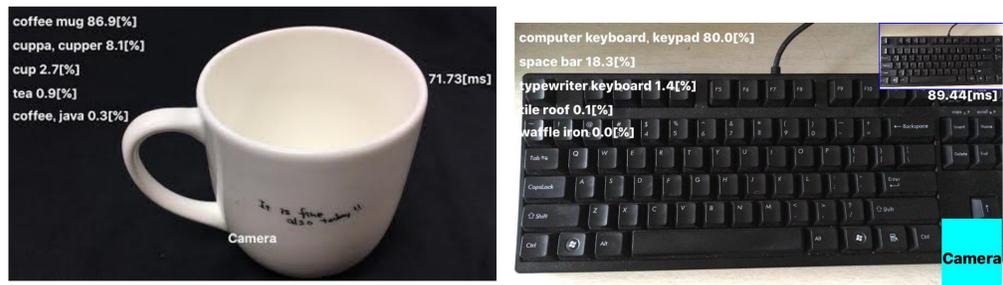
4. DeepDogCam



- ・犬100種類認識アプリ
- ・CNN学習: 4324枚(Stanford Dogs Dataset)

認識精度	Top-1	Top-5
犬100種類	69.0%	91.6%

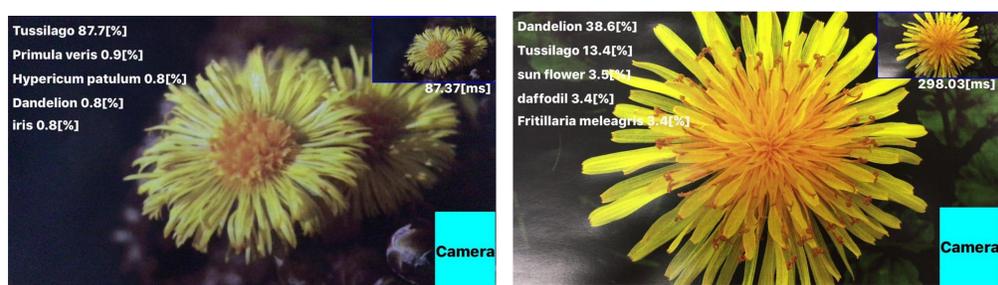
2. Deep2000Cam



- ・一般物体2000種類認識アプリ (ILSVRC1000種+ImageNet食事1000種類)
- ・CNN学習: 210万枚(ILSVRC1000+ImageNet1000)

認識精度	Top-1	Top-5
一般物体2000種類	39.8%	65.0%

5. DeepFlowerCam



- ・花17種類認識アプリ
- ・CNN学習: 1クラス80枚(17 Category Flower Dataset)

認識精度	Top-1	Top-5
花17種類	93.5%	99.1%

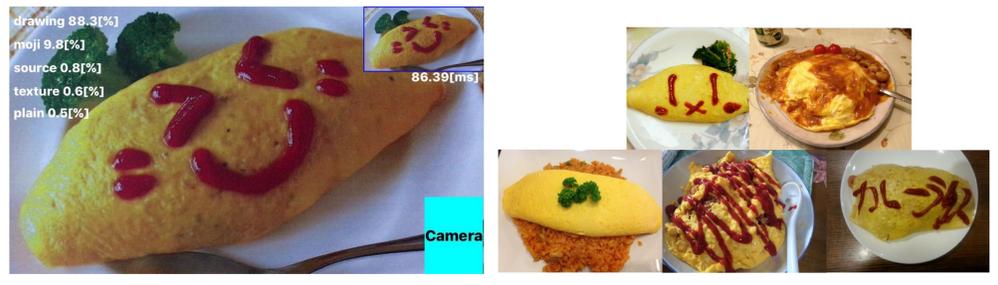
3. DeepBirdCam



- ・鳥200種類認識アプリ
- ・CNN学習: 6033枚(Caltech-UCSD Birds 200)

認識精度	Top-1	Top-5
鳥200種類	55.8%	80.2%

6. DeepOmeletCam



- ・オムレツ5種類認識アプリ(文字, 絵, 模様, ソース, プレーン)
- ・CNN学習: 各1万枚(プレーンのみ2400枚)

認識精度	Top-1	Top-5
オムレツ5種類	84.4%	-

7. DeepStyleCam

- ・Neural Style Transfer(スタイル変換)

—画像のコンテンツ(形状)を保持したまま, スタイル(テクスチャ)のみを変化

変換時間(250x250): 約180ms (iPad Pro)
約200ms (iPhoneSE)

