

Pregled programskih orodij za globoko učenje z vidika uporabe v industrijskih aplikacijah

Domen Tabernik, Danijel Skočaj

Univerza v Ljubljani

Fakulteta za računalništvo in informatiko

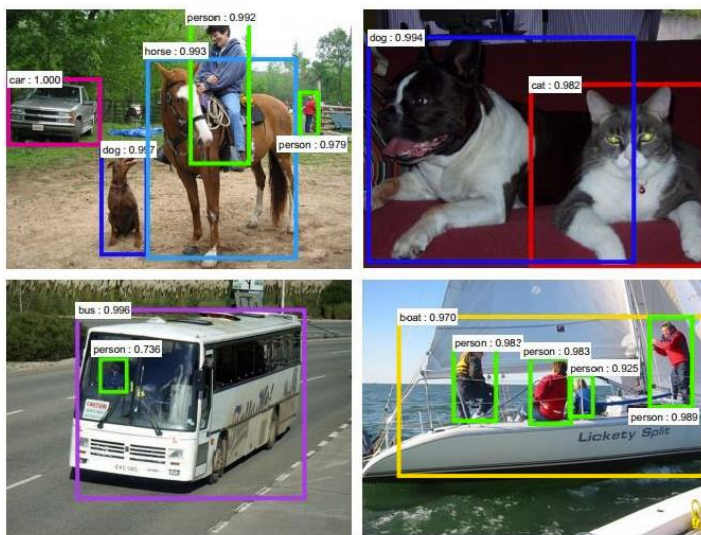
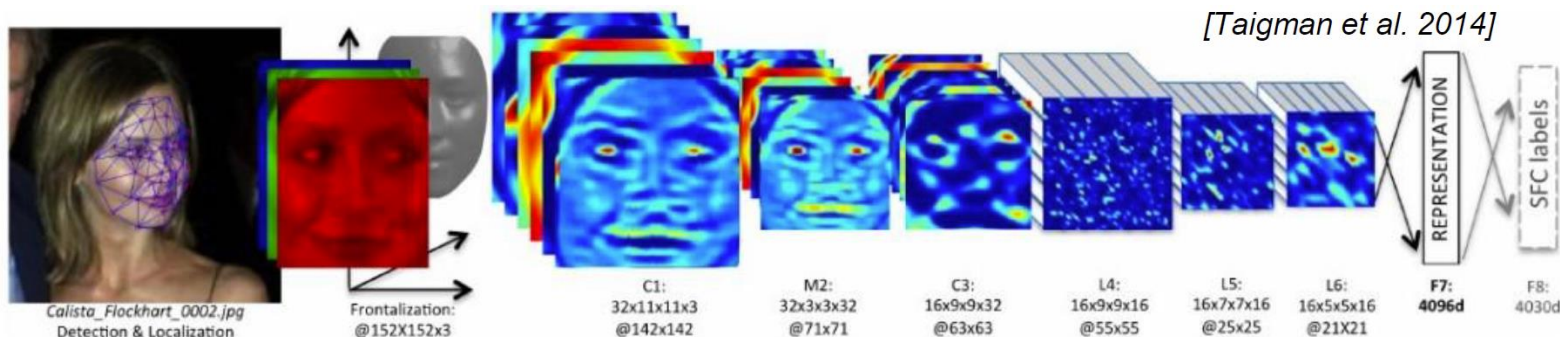
Laboratorij za umetne vizualne spoznavne sisteme



University of Ljubljana
Faculty of Computer
and Information Science

ViCOs
sua**l**gnitive
y**s**temslab

- Postaja pomembno orodje za reševanje problemov rač. vida



[Faster R-CNN: Ren, He, Girshick, Sun 2015]



[Farabet et al. 2012]

Orodje	Programski jeziki	Operacijski sistem	Licenca
<i>Caffe</i>	C/C++ in CUDA jedro, C++, Python, Matlab vmesnik	Primarno Linux, Windows slabše podprt	BSD 2-Clause
<i>Torch</i>	C/C++ in CUDA jedro, Lua vmesnik, Python	Primarno Linux, Windows slabše podprt	BSD 3-Clause
<i>Theano</i>	Python, C in CUDA jedro	Primarno Linux, Windows slabše podprt	BSD
<i>MatConvNet</i>	Matlab z Mex/C oz. CUDA jedri	Linux, Windows	Permissive BSD
<i>TensorFlow</i>	C/C++ in CUDA jedro, Python/C++ vmesnik	Primarno Linux, MacOS, Windows	Apache 2.0
<i>Keras</i>	Python vmesnik nad orodjem Theano ali TensorFlow	Odvisno od izbire jedra (Theano ali TensorFlow)	MIT

- Veliko preostalih knjižnic
- Nas zanimajo v kontekstu razvoja industrijskih aplikacij:
 - Enostavnost razvoja metod (pripravljena vizualizacijska orodja, itd.)
 - Enostavnost uporabe v produkciji (integracija v aplikacije, proces prevajanja, itd.)

- Prvo izmed modularnih orodij za ConvNet/CNN
- Razvija in podpira Berkeley Artificial Intelligence Research
- Najnovejši modeli običajno implementirani na Caffe:
 - (Faster) R-CNN, BatchNorm, ReLU, PReLU, Dropout
 - V zadnjem času tudi rekurzivne mreže (LSTM)
- Ogromna zbirka arhitektur in pred-naučenih modelov:
 - <https://github.com/BVLC/caffe/wiki/Model-Zoo>
 - VGG, NiN, ResNet, Places-CNN, FCNs, GoogLeNet, SegNet, ... (30+ state-of-the art modelov)

Implementacijski vidik:

- **Jezik:** C++ (Jedro) in C++/Matlab/Python (API)
- **OS:** primarno Linux, Windows slabše podprt
- **GPU podpora:** CUDA/CuDNN, eno-serverska multi-GPU podpora
- **Vizualizacijska orodja:** NE, uporabi Matlab oz. Python orodja

Integracijski vidik:

- Kar nekaj 3th-party knjižnic: *protobuf*, *glog*, *gflags*, *hdf5*, *ATLAS/OpenBLAS*, *Boost* in *CUDA*
- Problem *glog* in *gflags*:
 - Globalni knjižnici
 - Težko iztrgati jedro za integracijo v nove aplikacije

- Razvija in podpira Facebook-AI (FAIR)
- Splošno namenska knjižnica za globoko učenje, kot tudi za strojno učenje
 - ConvNets, RNN, LSTM, Restricted Boltzman Machine, itd.
 - K-Means, SVM, PCA, LDA itd
- Gradi na tenzorskih operacijah (ND-matrike)
 - Učinkovita implementacija CPU in GPU
 - Fleksibilnost pri gradnji novih mrež
- Veliko že pred-naučenih modelov
 - neuradni pretvorniki iz Caffe modelov v Torch modele (le za standardne ConvNet nivoje)

Implementacijski vidik:

- **Jezik:** C (Jedro) in Lua (API), poenostavljen MATLAB API in PyTorch
- **OS:** primarno Linux, neuradna Windows podpora iz skupnosti
- **GPU podpora:** CUDA/CuDNN tudi OpenCL, eno-serverska multi-GPU podpora
- **Vizualizacijska orodja:** NE, uporabi knjižnice v Lua

Integracijski vidik:

- Kompleksna integracija z veliko modularnih paketov
- Kar nekaj 3th-party knjižnic že v osnovnem paketu: *OpenBLAS*, *libjpeg*, *imagemagick*, *zeromq*, *graphicsmagick* in *xquartz*
- Ročno narejen skripte za prevajanje!

- Splošno namenska knjižnica za simbolično reševanje matematičnih optimizacijskih problemov
 - Idealen za probleme modelirane z računskim grafom
 - Iz simbolične definicije prevede problem neposredno v kodo C
 - Močno optimizirana koda za CPU in GPU (izkoristi tudi CuDNN)
- Model računskega grafa idealen za globoke mreže:
 - ConvNet, RNN, LSTM, itd
- Vendar, brez vnaprej pripravljenih standardnih globokih nivojev
 - Potrebno ročno ustvariti z grafom !!
- Tudi slab nabor pred-naučenih modelov (neuradni pretvorniki iz modelov orodja Caffe)

Implementacijski vidik:

- **Jezik:** C (Jedro) in Python (API)
- **OS:** primarno Linux, neuradna Windows podpora iz skupnosti
- **GPU podpora:** CUDA/CuDNN, eno-serverska multi-GPU podpora
- **Vizualizacijska orodja:** NE, uporabi knjižnice za Python

Integracijski vidik:

- API ne podpira C/C++
- Težja integracija v ločene knjižnice

- Konvolucijske mreže (ConvNet) kot programski paket (ang. toolbox) za orodje MATLAB
 - Konvolucijske in polno-povezane mreže
- Podpora za nekaj najbolj popularnih globokih nivojev
 - BatchNorm, ReLU, Dropout in ROI Pooling (za RCNN)
 - Brez podpore za rekurzivne mreže
- Nekaj pred-naučenih modelov:
 - AlexNet, VGG, GoogLeNet in ResNet
 - Slabša podpora za nove modele
 - Ni najnovejših modelov za semantično segmentacijo
 - Oxford skupina (pod vodstvom A. Vedaldi) ročno prekopira modele iz drugih orodij

Implementacijski vidik:

- **Jezik:** MEX-C (Jedro) in MATLAB (API)
- **OS:** Linux in Windows
- **GPU podpora:** CUDA na podlagi MATLAB implementacije (tudi CuDNN)
- **Vizualizacijska orodja:** NE, uporabi MATLAB orodja

Integracijski vidik:

- Integracija le skozi MATLAB !!

- Orodje za reševanje matematičnih problemov definiranih z računskimi grafi (podobno Theano)
 - Probleme modelira z računskim grafom → samodejno računa odvode in gradiente
- Pripravljeni različni globoki nivoji:
 - Konvolucije, max pooling, ReLU, Dropout, BatchNorm
- Neuradna podpora za druge metode (tf.contrib):
 - Rekurzivne mreže (RRN, LSTM), Monte Carlo, variacijske metode, CRF
- Pred-naučeni modeli
 - VGG, ResNet, Inception, autoenkoderji (<https://github.com/tensorflow/models>)
 - Neuradni pretvorniki iz Caffe modelov (le za standardne ConvNet nivoje, VGG, ResNet, GoogLeNet, NiN, AlexNet)

Implementacijski vidik:

- **Jezik:** C++ (Jedro) in C++/Python (API)
- **OS:** Linux, MacOS in Windows (le Python3.5)
- **GPU podpora:** CUDA/CuDNN, porazdeljena multi-GPU podpora !
- **Vizualizacijska orodja:** TensorBoard

Integracijski vidik:

- 3th-party knjižnice: *Python, six, NumPy, SciPy, protobuf, werkzeug in wheel* (Eigen - interno prevajanje)
- Avtomatiziran proces prevajanja: *Python PIP, bazel in conda*, (Cmake – neuradno)

Možnost uporabe pred-naučenih modelov le iz C++ API → odpadejo Python knjižnice !

- Programski ovojnica nad obstoječim orodjem za računske grafe (Theano in TensorFlow)
- Poenostavljen Python vmesnik
- Pripravljeni obstoječi globoki nivoji:
 - konvolucijski nivo, polno povezani nivo, rekurzivni nivo, BatchNorm, ReLU, PReLU, itd
- Pred-naučeni modeli
 - VGG, ResNet, Inception
 - Neuradni pretvorniki iz Caffe modelov (le za standardne ConvNet nivoje)
 - Ne podpira neposredne uporabe modelov TensorFlow !

Implementacijski vidik:

- **Jezik:** Python (API)
- **OS:** Odvisno od izbire jedra
- **GPU podpora:** Odvisno od izbire jedra
- **Vizualizacijska orodja:** Odvisno od izbire jedra

Integracijski vidik:

- Uporaba le preko Python
- Pri uporabi TensorFlow lahko izvozimo v TensorFlow modele

→ *Uporaba TensorFlow C++ jedra/vmesnika*

- Izbira odvisna od specifičnih zahtev
 - Z raziskovalnega vidika
 - Z integracijskega vidika industrijskih aplikacij
- Uporaba za raziskovalce:
 - Za osnovne arhitekture
 - MatConvNet
 - Za aktualne arhitekture
 - Caffe (težavnost implementacije novih nivojev)
 - Torch (Lua vmesnik!)
 - TensorFlow
 - Theano (+Keras)
- Uporaba v industrijskih aplikacijah
 - Caffe (težavnost integracije kot ločena knjižnica)
 - TensorFlow (+Keras)

→ Za naše potrebe izbrali TensorFlow