



Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES



inovex

AI Labor - Sommersemester 2019

Computer Vision
3. Termin

Robin Baumann, Stanislav Frolov

Karlsruhe, 05. April 2019

Agenda für Heute

- Datensatz CIFAR-10
- Praktischer Teil: Bildklassifizierung
 - MLP
 - TensorBoard
 - CNN

Datensatz CIFAR-10

Allgemeines zu CIFAR-10

Canadian Institute for Advanced Research (CIFAR)

- 60k RGB Bilder
- 32x32 Pixel
- 10 Klassen
 - 6k pro Klasse
 - airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck
- 50k train, 10k test
- CIFAR-10 und CIFAR-10 are labeled subsets of the [80 million tiny images](#) dataset

airplane



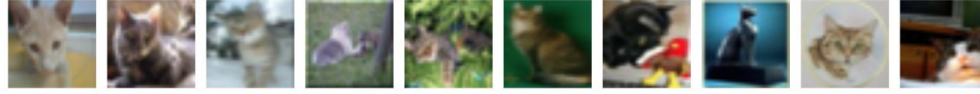
automobile



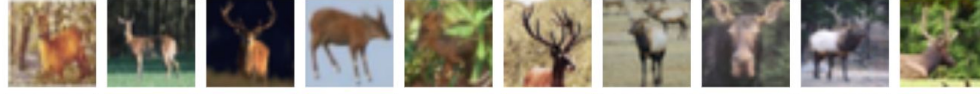
bird



cat



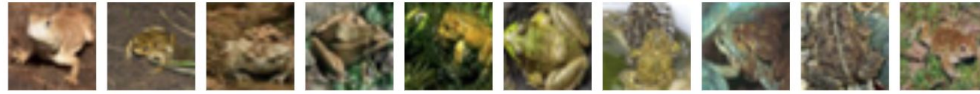
deer



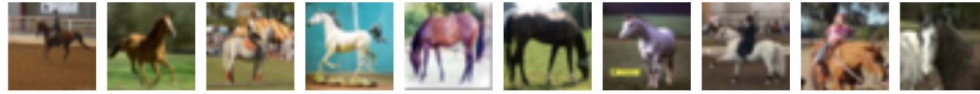
dog



frog



horse



ship



truck



Struktur des Datensatzes

- Aufgeteilt in 6 batches (je 10k Bilder):
 - 5 train batches
 - 1 test batch
- test batch besteht aus jeweils 1k zufällig ausgewählten Bildern pro Klasse
- Jeder batch wird als [pickle](#) File geliefert

Struktur des Datensatzes

- Jeder batch besteht aus einem *dictionary* mit:
 - data: 10000x3072 *numpy array* aus `uint8`
 - *jede Zeile speichert ein 32x32 RGB Bild*
 - *[1024 rot, 1024 grün, 1024 blau] Einträge*
 - *die ersten 32 Einträge sind die Rotwerte der ersten Zeile im Bild*
 - labels: Liste aus 10000 Nummern im Bereich 0-9
 - *Nummer am Index i steht für das Label des Bildes i im data array*

Struktur des Datensatzes

- Zusätzlich gibt es noch eine File *batches.metadata*
 - Enthält ein *dictionary* mit den Label-Namen, z.B.:
 - `label_names[0] == "airplane"`
 - `label_names[1] == "automobile"`

Research Paper	Error rate (%)	Publication Date
Convolutional Deep Belief Networks on CIFAR-10 ^[6]	21.1	August, 2010
Densely Connected Convolutional Networks ^[7]	5.19	August 24, 2016
Wide Residual Networks ^[8]	4.0	May 23, 2016
Neural Architecture Search with Reinforcement Learning ^[9]	3.65	November 4, 2016
Fractional Max-Pooling ^[10]	3.47	December 18, 2014
Shake-Shake regularization ^[11]	2.86	May 21, 2017
Coupled Ensembles of Neural Networks ^[12]	2.68	September 18, 2017
ShakeDrop regularization ^[13]	2.67	Feb 7, 2018
Improved Regularization of Convolutional Neural Networks with Cutout ^[14]	2.56	Aug 15, 2017
Regularized Evolution for Image Classifier Architecture Search ^[15]	2.13	Feb 6, 2018
AutoAugment: Learning Augmentation Policies from Data ^[16]	1.48	May 24, 2018

Praktischer Teil

Aufgaben

TODOs implementieren:

1. MLP trainieren (Ziel: >40% test accuracy)
 - › flatten, dense, dropout, ...
2. TensorBoard summaries für train und test
3. CNN trainieren (Ziel: >70% test accuracy)
 - › flatten, dense, dropout, conv2d, max_pooling2d, ...

Setup

- Anmeldung am Rechner
- Ethernet auf `hskaopen` umstellen
- Wired Settings:
 - `testXYZ` entfernen und mit Studentenkürzel ersetzen
 - Passwort eingeben

Setup

- Terminal öffnen

- `git clone`
`https://github.com/hskaailabcv/source.git`
- `cd source`
- `docker-compose up`

- Jupyter: <http://localhost:8888>

Feedback

- Google Forms

<https://forms.gle/5yafxDPP1beLQ4g2A>



Vielen Dank

Robin Baumann

rbaumann@inovex.de

Stanislav Frolov

sfrolov@inovex.de

inovex GmbH

Ludwig-Erhard-Allee 6

76131 Karlsruhe



