

III. Physikalisches
Institut A

RWTHAACHEN
UNIVERSITY

VISPA Tutorial

*Martin Erdmann, Peter Fackeldey, Benjamin Fischer, **Jonas Glombitza**, Dennis Noll, Thorben Quast, Yannik Rath, Marcel Rieger, Marcus Wirtz*

III. Physikalisches Institut A, RWTH Aachen

SPONSORED BY THE



HTTPS

VISPA

SSH



Federal Ministry
of Education
and Research

The VISPA Project



Software

Access via Web Browser

- ~ 100 active users
- Large updated software environment
 - ♦ Python, numpy, matplotlib, TensorFlow, ROOT etc.
- Physics specialized extensions

Cluster

Working environment

- ~200 CPU Cores
- 20 GPUs
- HTCondor



Opens the example page

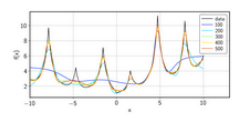


- Developed in Aachen (group of Prof. Erdmann)
- GPU extension
 - ♦ 20 NVIDIA GTX 1080
- Accessible via <https://vispa.physik.rwth-aachen.de/>





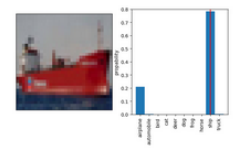
Deep Learning



1D Function Fitting

In this example, you can train a neural network to fit an arbitrary function and investigate the approximation performance during the training iterations.

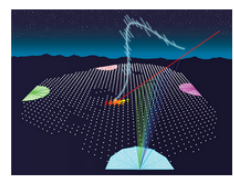
[Reset example](#) [Open example](#)



CIFAR-10 Image Classification

CIFAR-10 is a dataset of tiny natural images showing objects of 10 different classes. It is a popular data set for experimenting with different deep learning techniques. In the provided examples you can train and apply: a fully connected net, a simple convolutional net and a deep convolutional net.

[Open example](#)

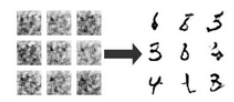


Deep learning based Air Shower Reconstruction

Ultra-high energy cosmic rays produce extensive air showers of secondary particles upon entering the atmosphere. Sampling the footprint of these particles with surface detectors is a widely used detection technique. In this example you can exploit advanced convolutional techniques to reconstruct the energy, showeraxis and depth of the shower maximum of cosmic ray induced air showers.

[Open example](#)

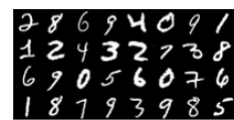
Deep Generative Models



Generative Adversarial Networks (GANs) for MNIST

In this example, you can generate handwritten digits by training a Deep Convolutional Generative Adversarial Network (DCGAN) to the MNIST data set.

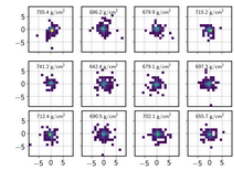
[Reset example](#) [Open example](#)



MNIST Digit Recognition

MNIST is a dataset of 28x28 greyscale images of handwritten digits and a classic task for benchmarking image classification algorithms. In this example, you can train and apply a simple convolutional neural network to identify the correct digit.

[Reset example](#) [Open example](#)

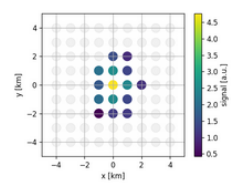


Air Shower Classification

Ultra-high energy cosmic rays produce extensive air showers, which vary among others with the cosmic ray mass. For tracking the cosmic rays back to their sources, the reconstruction of the charge is a key parameter hence it could allow for estimating the galactic magnetic field deflection for each cosmic ray. In this example you can train a neural network on a toy data set to reconstruct the mass of the air shower (classification) and the showermaximum (regression).

[Open example](#)

Open Air Shower example



Wasserstein GANs for Physics Simulations

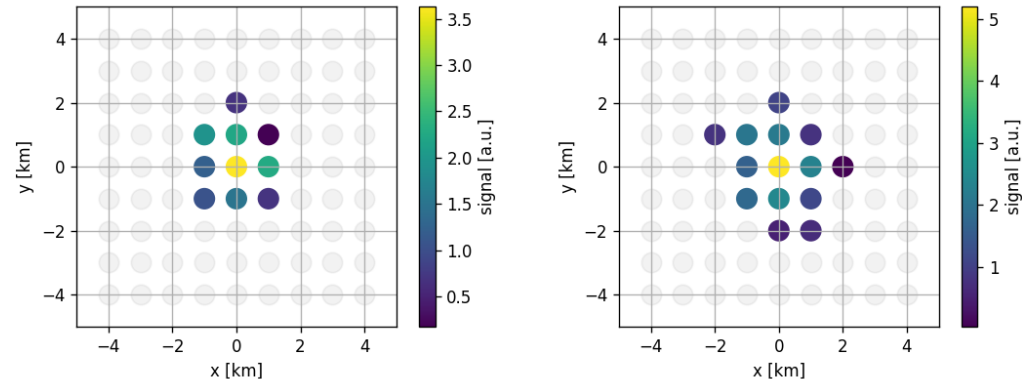
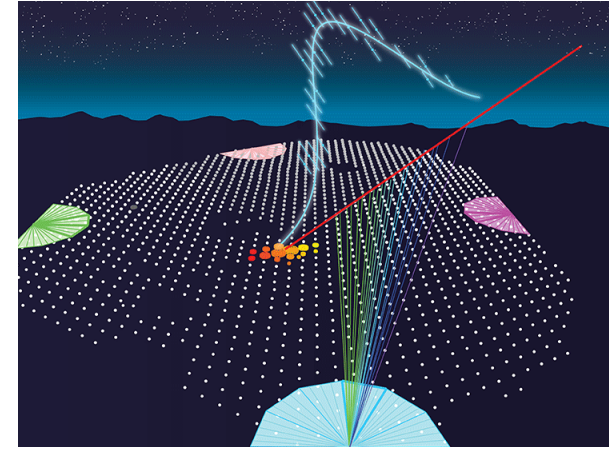
In this example, you can train a improved Wasserstein Generative Adversarial Network (WGAN) to generate signal patterns of cosmic ray induced air showers.

[Reset example](#) [Open example](#)

Astroparticle Examples

Air Shower Reconstruction

- Cosmic Ray induced Air Showers
 - ♦ Pierre Auger like Observatory
- Reconstruct energy and shower axis
- Use footprint as 2D image
 - ♦ Center image at hottest tank
- Use arrival times and signals
 - Resulting shape 9 x 9 x 2





tools.py
DCNN_axis.py
CNN_energy.py

Open CNN-energy.py

Type: pygpu '%file'

and click execute button → submit job to GPU cluster



Examples x MNIST_GAN x MNIST_GAN.py x \$HOME/MNIST/mni... x mnist_cnn.py x AirShower_WGAN x ShowerGAN.py x airshower_auger x CNN_energy.py x

Save x Close Save as ...

```
1'''
2Advanced Convolutional network for the Air Shower reconstruction at an "Auger like observatory".
3Run this script with 'pygpu %file' in the code editor or 'pygpu DNN_Energy.py' in the terminal.
4https://doi.org/10.1016/j.astropartphys.2017.10.006
5'''
6import numpy as np
7import os
8import sys
9import tools
10import tensorflow as tf
11from tensorflow import keras
12models = keras.models
13layers = keras.layers
14
15# -----
16# Boilerplate. You can ignore this part.
17# -----
18# -----
19try:
20    CONDIR_ID = os.environ['CONDOR_ID']
21except KeyError:
22    sys.exit('Error: Run this script with "pygpu %file"')
23tf.logging.set_verbosity(tf.logging.ERROR)
24
25folder = 'train-Energy-Xs-%CONDOR_ID # folder for training results
26os.makedirs(folder)
27
28# -----
29# Load and prepare dataset - you can leave as is.
30# -----
31data = np.load('/net/scratch/deepLearning/airshower/auger-shower-planar.npz')
32
33# time map, values standard normalized with untriggered stations set to 0
34T = data['time']
35T -= np.nanmean(T)
36T /= np.nanstd(T)
37T[np.isnan(T)] = 0
38
39# signal map, values normalized to range 0-1, untriggered stations set to 0
40S = data['signal']
41S = np.log10(S)
42S = np.nanmin(S)
43S /= np.nanmax(S)
44S[np.isnan(S)] = 0
45
46# Input Features
47X = np.stack([T, S], axis=-1)
48
49# energy - E in EeV [0-100 EeV]
50logE = data['logE']
51energy = 10**(logE - 18)
52
53
```

Execute pygpu '%file'

3/27/2019, 2:57:30 PM:
executing "pygpu 'CNN_energy.py'"

OUTPUT:
Require GPUs: 2
Require Memory: 4096
Require CPUs: 1
CondorID: 372495.0
Waiting to be executed...
Executing on slot2@vispa-gpu08.physik.rwth-aachen.de"

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 9, 9, 16)	304
conv2d_2 (Conv2D)	(None, 9, 9, 16)	2320
max_pooling2d_1 (MaxPooling2)	(None, 4, 4, 16)	0
conv2d_3 (Conv2D)	(None, 4, 4, 32)	4640
conv2d_4 (Conv2D)	(None, 4, 4, 32)	9248
global_max_pooling2d_1 (Global	(None, 32)	0

/home/KoenigAlfonsDerViertelVorElfte/airshower_auger

Results – Energy Reconstruction



III. Physikalisches
Institut A

RWTHAACHEN
UNIVERSITY

```

y
x
AirShower_WGAN
ShowerGAN.py
airshower_auger
CNN_energy.py
+

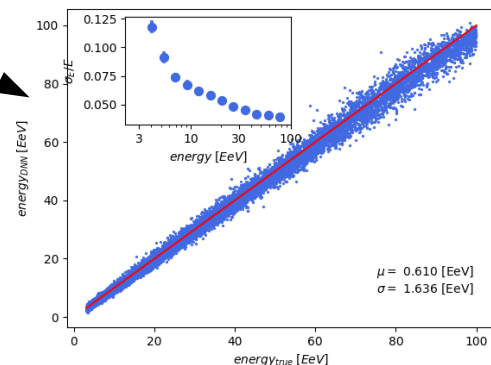
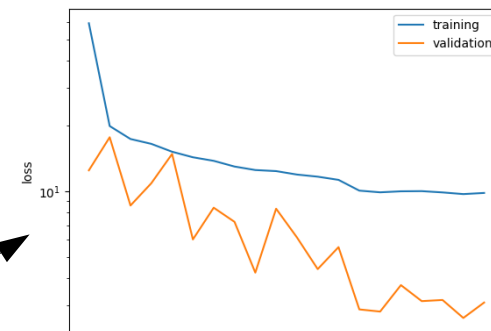
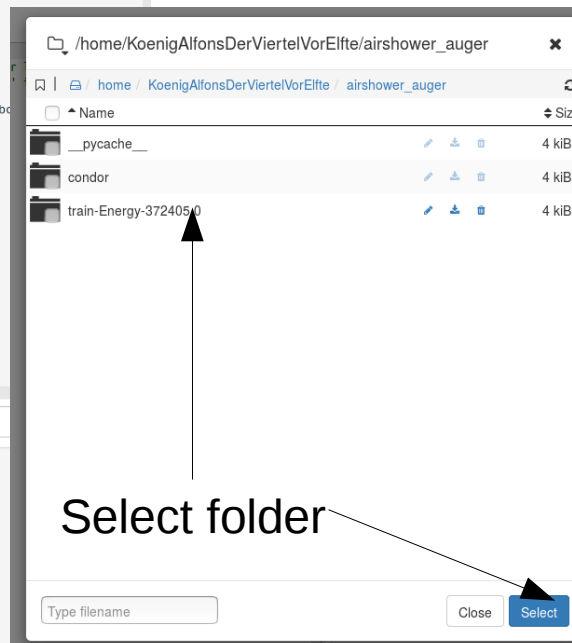
Execute
pygpu '%file'

- 4s - loss: 34.5520 - Distance: 5.7111 - val_loss: 11.6223 - val_Distance: 2.6923
Epoch 13/20
- 4s - loss: 34.2457 - Distance: 5.6790 - val_loss: 4.4938 - val_Distance: 2.0929
Epoch 14/20
- 4s - loss: 33.1566 - Distance: 5.5915 - val_loss: 5.2250 - val_Distance: 2.2271
Epoch 15/20
- 4s - loss: 32.1196 - Distance: 5.5133 - val_loss: 7.8445 - val_Distance: 2.5696
Epoch 16/20
- 4s - loss: 31.6609 - Distance: 5.4675 - val_loss: 9.9845 - val_Distance: 2.8606
/software/python3.5/lib/python3.5/site-packages/seaborn/apionly.py:6: UserWarning: As seaborn
warnings.warn(msg, UserWarning)
Epoch 17/20
- 4s - loss: 30.6934 - Distance: 5.3933 - val_loss: 4.1746 - val_Distance: 1.9724
Epoch 18/20
- 4s - loss: 30.5457 - Distance: 5.3796 - val_loss: 4.7332 - val_Distance: 2.0065
Epoch 19/20
- 4s - loss: 30.2052 - Distance: 5.3398 - val_loss: 7.6627 - val_Distance: 2.3834
Epoch 20/20
- 4s - loss: 29.4051 - Distance: 5.2675 - val_loss: 4.8603 - val_Distance: 2.1747
mean -0.11504151512499707 std 2.1725086751282343

runtime: 1 m, 39.01 s
=====

/home/KoenigAlfonsDerViertelVorElfte/airshower_auger
Click folder icon

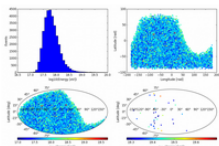
```



More Examples

- Feel free to try out more applications
 - ♦ Generative Models
 - ♦ Astroparticle Applications

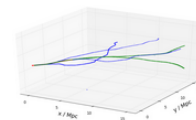
Astroparticle Examples



Pierre Auger Observatory Data Analysis Example

In this example, you can use the public data set (1 % of all events) of the Pierre Auger Observatory to study the energy distribution and to plot the arrival directions of cosmic rays.

[Open example](#)

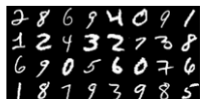


CRPropa Example

In this example, you can use CRPropa to simulate the propagation of a cosmic ray nucleus in a turbulent extragalactic magnetic field.

[Open example](#)

Generative Adversarial Networks (GANs)



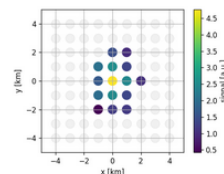
MNIST Digit Recognition

MNIST is a dataset of 28x28 greyscale images of handwritten digits and a classic task for benchmarking image classification algorithms. In this example, you can train and apply a simple convolutional neural network to identify the correct digit.

[Reset example](#)

[Open example](#)

Wasserstein GANs



Wasserstein GANs for Physics Simulations

In this example, you can train a improved Wasserstein Generative Adversarial Network (WGAN) to generate signal patterns of cosmic ray induced air showers.

[Reset example](#)

[Open example](#)

- Generate MNIST digits

- Generate Air Shower Footprints

The VISPA Platform

- Accessible with web browser – GPU extension
 - ♦ Many software packages available
- Connection from VISPA to my own preferred computing resources

VISPA as Marketplace?

- Exchange of code, architectures, training methods
 - ♦ Direct execution / testing on cluster
- Exchange of pre-trained models

