

Thesis for obtaining the academic degree
Bachelor of Science

Finding optimal hyperparameters for cleaning
algorithms for the Cherenkov Telescope
Array

Anno Knierim
born in Hagen

2022

Chair of Experimental Physics V

Department of Physics

TECHNISCHE UNIVERSITÄT DORTMUND

Reviewer: Prof. Dr. Dr. Wolfgang Rhode
Co-reviewer: Dr. Dirk Wiedner
Submission date: 7 September 2022

This thesis is set in Libertinus (Serif, Sans and Math) and Fira Code typeset using L^AT_EX with LuaT_EX from T_EXLive 2021.
Title graphic by **TITLE GRAPHIC AUTHOR**.

Abstract

The abstract is a short summary of the thesis in English, and together with the German summary, it has to fit on this page.

Kurzfassung

Hier steht eine Kurzfassung der Arbeit in deutscher Sprache inklusive der Zusammenfassung der Ergebnisse. Zusammen mit der englischen Zusammenfassung muss sie auf diese Seite passen.

Todo list

rewrite this (maybe)	1
confirm values and add source	4
wording	4
Adjust figure size of Figure 2.2	5
maybe write something about ML	5
decide on final figure, check source for south	6
full or alpha	6
decide on Figure 2.4	6
write about camera tech	6
write about scientific goals for the cta	6
citation	9
find alternative wording	10
wording	10
think about connections between chapter 4 and ??	10
watch out for text boundaries	13

Contents

1	Gamma-Ray Astronomy	1
2	IACTs and the Cherenkov Telescope Array	4
2.1	Imaging Air Cherenkov Telescopes	4
2.2	The Cherenkov Telescope Array	5
3	Data processing with <code>ctapipe</code>	8
3.1	Data Levels in <code>ctapipe</code>	8
3.2	This work's data processing pipeline	10
4	Finding Optimal Hyperparameters for the Cleaning Algorithms	11
4.1	Cleaning Algorithms	11
4.2	Hyperparameters	13
5	Results	16
6	Conclusions and Outlook	17
Bibliography		18
Glossary		20
Appendix		21
1	Configurations for <code>ctapipe</code>	21
Acknowledgements		25

Gamma-Ray Astronomy

Astronomy, being one of the oldest sciences, is a vast field of study dating back to the Babylonians. From the earliest days of civilization, astronomers have been studying the stars and the planets to understand the universe. It, therefore, is no surprise that astronomy spawned a great number of discoveries throughout the centuries. Whereas first observations were made by eye only, we now have access to a multitude of experiments and telescopes that deepen our understanding of the universe. With the discovery of cosmic rays (CR) by Victor Hess in the early 20th century, the new field of astroparticle physics was born [12].

From then on we found many different types of cosmic messengers, the most recent being the discovery of gravitational waves in 2015 [1].

rewrite this
(maybe)

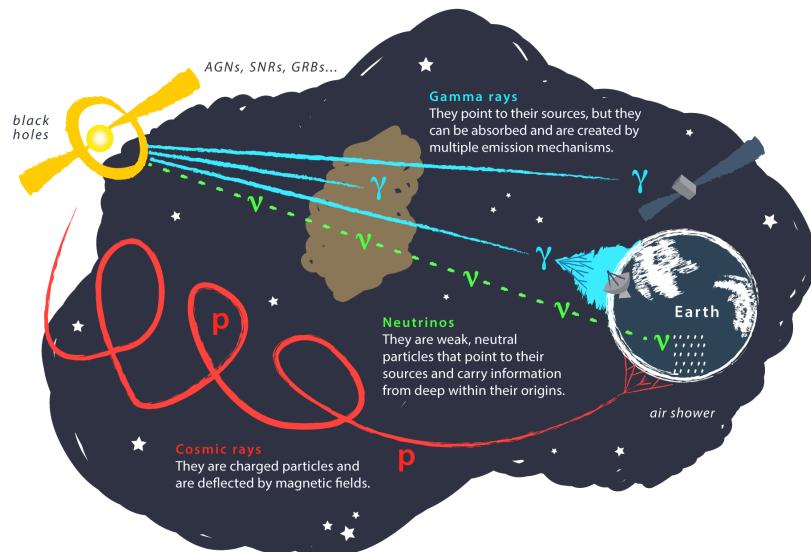


Figure 1.1: Different types of cosmic rays on their way to Earth. Charged particles like protons and electrons are deflected by magnetic fields and therefore making it hard to pinpoint the source. Only the origin of photons and neutrinos can be reconstructed directly since they are uncharged particles and therefore travel in straight lines. However, photons can be absorbed or created in multiple mechanisms. Since neutrinos only rarely interact with matter via the weak force, their detection is significantly harder than for photons [3].

CR come in different types, that are either charged or uncharged, as shown in Figure 1.1. Charged particles like electrons, protons or atomic nuclei are difficult to trace back to their origin as they are deflected by the cosmic electromagnetic fields. Uncharged particles like photons or neutrinos, however, travel in straight lines, making it easier to reconstruct their origins, although photons can be absorbed by dust clouds in their way.

Since neutrinos are harder to detect due to their weak interaction with matter, photons are easier to study with space- and ground-based experiments.

Therefore, in recent years, gamma-ray astronomy has become an important research field in astroparticle physics. The term gamma-rays is generally denoted as photons with energies above 100 keV [9]. Due to this high-energy nature, gamma rays pose some of the most powerful CR in the universe and since photons at such energies cannot be produced by thermal processes, their origin can be described by higher order processes involving charged particles. Sources for gamma rays include supernova remnants (SNRs), Pulsars, Blazars, active galactic nuclei (AGNs) or gamma ray bursts (GRBs).

SNRs are the remnants of a supernova explosion.

Pulsars are rotating neutron stars that emit gamma rays in the form of pulsations.

Blazars are AGNs that are highly variable and emit gamma rays in the form of flares.

AGNs are active galactic nuclei that emit gamma rays in the form of flares.

GRBs are gamma-ray bursts that are the most energetic explosions in the universe.

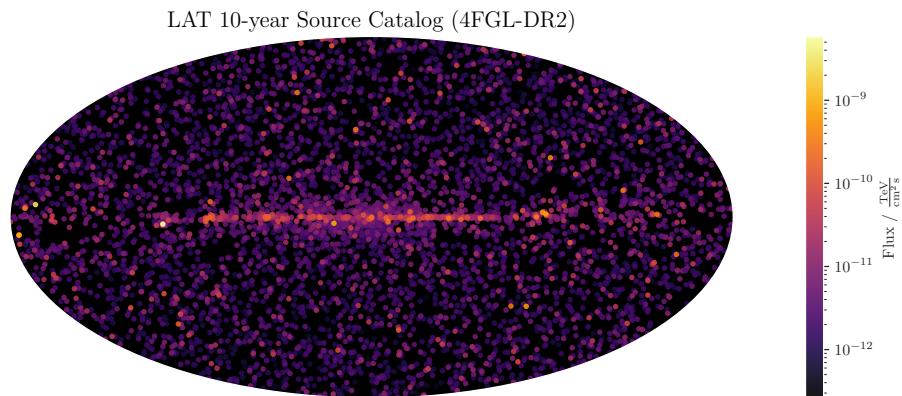


Figure 1.2: Hammer projection of the Fermi-LAT 4FGL catalog data of gamma-ray sources. The skymap shows the flux of gamma-ray sources in $\text{TeV}/(\text{cm}^2 \text{s})$ over a span of 10 years [2].

Gamma rays can be detected by a variety of experiments, either ground- or space-based. Space-based experiments like the Fermi Large Area Telescope (Fermi-LAT) are usually more sensitive to lower energies, whereas ground-based experiments are more sensitive to higher energies. The Fermi-LAT is a space-based gamma-ray observatory that was launched in 2008 and is still in operation today. Figure 1.2 shows the gamma-ray sky as observed by the Fermi-LAT over a span of 10 years.

For the past two decades, ground-based Imaging Air Cherenkov Telescopes (IACT) experiments like the Major Atmospheric Gamma-Ray Imaging Cherenkov (MAGIC) telescopes, the Very Energetic Radiation Imaging Telescope Array System (VERITAS) and the High Energy Stereoscopic System (H. E. S. S.) have

been monitoring these very-high-energy gamma rays (VHE gamma rays) to gain an understanding of their production. This allowed us to determine different source classes inside and outside our galaxy, with the most important source class inside our galaxy being SNRs such as the Crab Nebula.

IACTs and the Cherenkov Telescope Array

2

Most modern gamma-ray observations are performed with either space-based experiments or with Imaging Air Cherenkov Telescopes (IACTs), which are ground-based telescopes or arrays of telescopes that use the Cherenkov light emitted by extensive air showers (EASs) in the atmosphere. In the following sections I will introduce IACTs and the Cherenkov Telescope Array (CTA) and explain the mechanisms that make it possible to observe gamma rays with these types of experiments.

2.1 Imaging Air Cherenkov Telescopes

confirm values and add source

Because of their ground-based setup, IACTs are taking advantage of the Earth's atmosphere to get a larger effective area than any space-based instrument. This is especially helpful for energies above 100 GeV, where the gamma-ray flux is low compared to lower energies that have higher fluxes. This means, that space-based experiments with their small effective area will see fewer high-energy events compared to ground-based experiments. The cosmic ray flux in Figure 2.1 shows this well: The flux decreases rapidly with higher energies, with only one high-energy photon per day and square meter reaching Earth from the Crab Nebula [13] and even fewer photons for energies in the EeV domain, where only 1 particle per square kilometer per year reaches Earth.

wording

For high-energy gamma rays, Earth's atmosphere is opaque, so ground-based experiments rely on cascades of sub-particles in so-called EASs. For gamma-ray astronomy, the electromagnetic component of these EASs, shown in Figure 2.2a, is more relevant than the hadronic component. When a gamma ray interacts with Earth's atmosphere, it decays into an electron and a positron via pair production. These charged particles then emit more photons via bremsstrahlung. These photons, again, produce more charged particles, which in turn emit more photons. This process continues until any of these processes reaches energies below a certain threshold, i. e. 1022 keV for the pair production process. For these processes to happen, the particles have to be in the fields of the atoms of the Earth's atmosphere. As the particles travel at speeds faster than

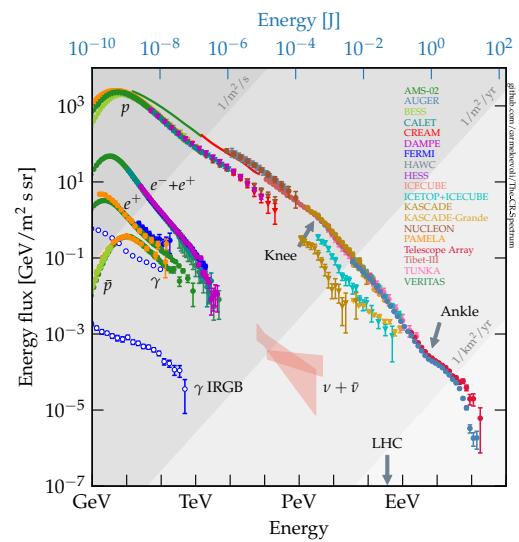
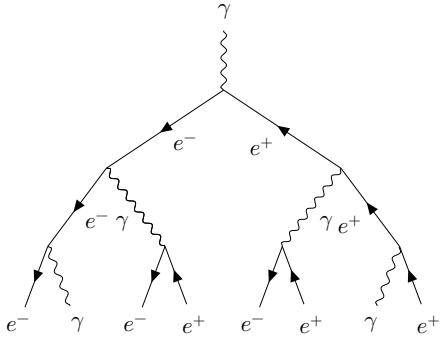


Figure 2.1: The cosmic ray flux as a function of energy. For very high energies the flux becomes very small with only 1 particle per square meter per year reaching Earth. For even higher energies in the domain of EeV this flux becomes 1 particle per square kilometer per year [8].

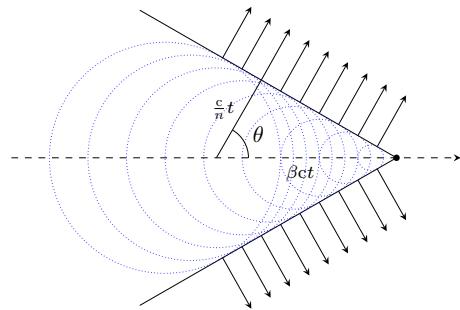
light in the medium of the atmosphere, they emit Cherenkov light at a fixed angle θ with respect to the refraction index n of the atmosphere and the factor $\beta = v/c$. The angle can be determined trigonometrically as

$$\cos \theta = \frac{1}{\beta n}, \quad (2.1)$$

as can be seen in Figure 2.2b.



(a) Simplified Heitler model of the electromagnetic component of an extensive air shower. A gamma ray induces an electron and a positron via pair production that then emit more photons via bremsstrahlung.



(b) Cherenkov radiation (outward-pointing arrows) from a charged particle traveling at uniform velocity, faster than the speed of light in the medium. The dashed line shows the path of the particle over time.

Figure 2.2: The electromagnetic component of an extensive air shower (see Figure 2.2a) is induced by gamma rays that then decay into electrons and positrons via pair production. These charged particles will emit more gamma rays via bremsstrahlung, which again will produce new electrons and positrons. This will continue until e.g. the energy of the gamma rays is below the threshold of 1022 keV for pair production. Another process is the production of Cherenkov light from the electrons and positrons, which is shown in Figure 2.2b. The Cherenkov light is the result of the charged particles traveling faster than the speed of light in the atmosphere, thus emitting photons in a fixed angle θ w.r.t. the refraction index n of the medium and the factor β .

Cherenkov light emitted by EASs can then be collected by an IACT's mirrors and detected by its camera within a timeframe of an order of nanoseconds. The resulting image will show the shape of the impact of the air shower and can be used to determine the shower's primary particles' properties, and the type and reconstruct its origin. As the hadronic component of EASs produce electromagnetic subshowers, IACTs have a dominant hadronic background, which has to be separated from the gamma ray-induced showers.

Adjust figure size of Figure 2.2

maybe write something about ML

2.2 The Cherenkov Telescope Array

The CTA is a new generation of IACTs that will consist of two sites, one of which will be built at the Observatorio del Roque de los Muchachos (ORM) on the Canarian island of La Palma while the other will be built in the southern hemisphere at the European Southern Observatorys (ESO) Paranal Observatory in the Atacama desert of northern Chile. In this work, I will focus on the northern array, also called CTA north, on La Palma.

The CTA consists of three types of telescopes, shown in Figure 2.3, each being sensitive to a different energy range [5]:

Large-Sized Telescope (LST): The largest telescopes in the CTA have primary deflector diameter of 23 m with a field of view of 4.3 deg and are sensitive to energy ranges from 20 GeV to 150 GeV.

Medium-Sized Telescope (MST): The MSTs have a primary deflector diameter of 11.5 m with a field of view of 7.7 deg for their NectarCam and are sensitive to energy ranges from 150 GeV to 5 TeV.

Small-Sized Telescope (SST): Being the smallest telescope type in the CTA, the SSTs have a primary deflector diameter of 4.3 m, a secondary deflector diameter of 1.8 m with a field of view of 10.5 deg and are sensitive to energy ranges from 5 TeV to 300 TeV.

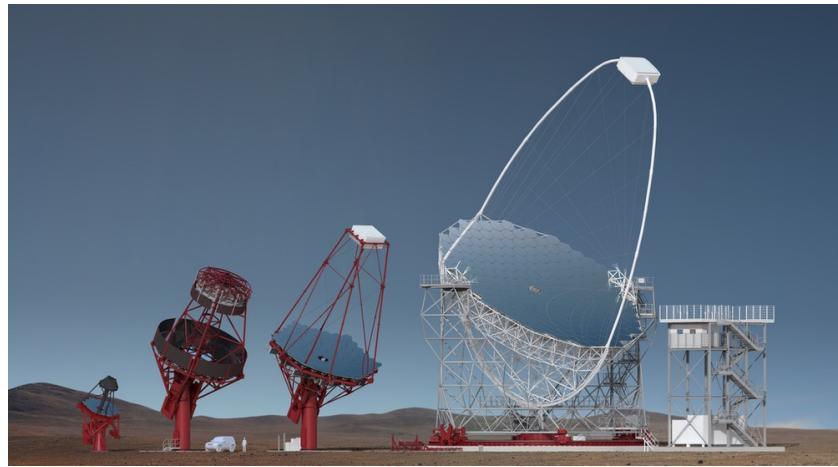


Figure 2.3: A rendered image of the 3 telescope prototypes in CTA. From left: The SST, two types of MST (SCT and MST) and the LST. This work will not feature the SCT, as the used simulation data contains only MST and MST data for the northern array. Image credit: Gabriel Pérez Diaz, IAC [15].

Since the southern hemisphere has more galactic high-energy sources than the northern hemisphere, only CTA south will feature the SSTs along the other two telescope types. CTA north will feature the MSTs and LSTs. Figure 2.4 shows the planned full layout of both sites of the CTA where CTA south is displayed with its 70 SSTs, 25 MSTs and the 4 LSTs in the center. CTA north will feature 15 MSTs and 4 LSTs. Due to the simulation data used, this work will focus on the reduced layout, also called Alpha Configuration, for the northern site, consisting of 9 MSTs and 4 LSTs.

decide on final figure, check source for south

full or alpha

decide on Figure 2.4

write about camera tech

write about scientific goals for the cta

The MST's and LST's cameras are based on photo multiplier tube (PMT) photodetectors, while the SST's camera is based on a silicon photomultiplier (SiPM) photodetector.

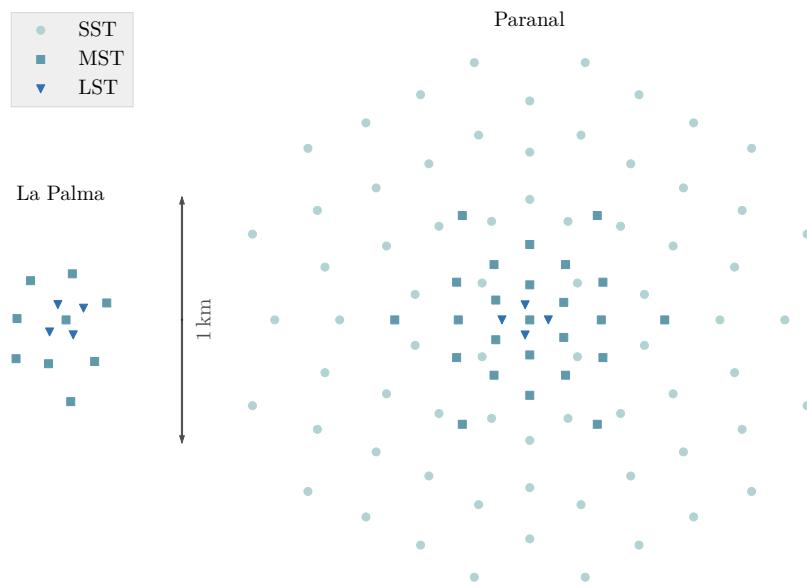


Figure 2.4: Alpha Configuration layout of the CTA north and full configuration of the CTA south site. CTA south would feature 70 SSTs, 25 MSTs and 4 LSTs [7] its full configuration, while CTA north would feature 9 MSTs and 4 LSTs in it's Alpha Configuration [6]. This figure was adapted from a figure by Kai Brügge in his Ph.D. thesis [4].

Data processing with ctapipe

3

The analysis in this work was done with the open-access low-level data processing software for CTA, `ctapipe` [11]. The version used is the development version `0.15.1.dev166+gf26107f`, henceforth shortened to version `0.15.1`. The goal of `ctapipe` is to provide a complete analysis framework ranging from data calibration and image extraction to the reconstruction of events and the analysis of their properties. In this chapter, I will describe `ctapipe` with its different data levels and the different analysis steps (section 3.1). In section 3.2 I will explain the full pipeline created for this work.

3.1 Data Levels in `ctapipe`

There are several data levels in `ctapipe`, spanning from the raw data **R0** to the reconstructed events **DL2**, with the raw data levels being denoted by a **R** and the calibrated data levels by a **D**. section 3.1 shows a simplified overview of the data levels and the analysis steps. The raw data level **R0** is the data that comes directly from the photodetectors. The time-resolved signal of the data is calibrated from **R0** to **R1**. Then the data volume gets reduced (**R1** → **DL0**) by a selection of waveforms. The **DL0** data level is the first level being stored and also the first level to be processed from the simulation datasets used in this work. From **DL0** to **DL1a**, the images of the data are extracted from the time pulses, which are then cleaned by a cleaning algorithm, allowing for parametrization of the events (**DL1a** → **DL1b**). The parametrized events can then be reconstructed (**DL1b** → **DL2**) and stored on the **DL2** data level.

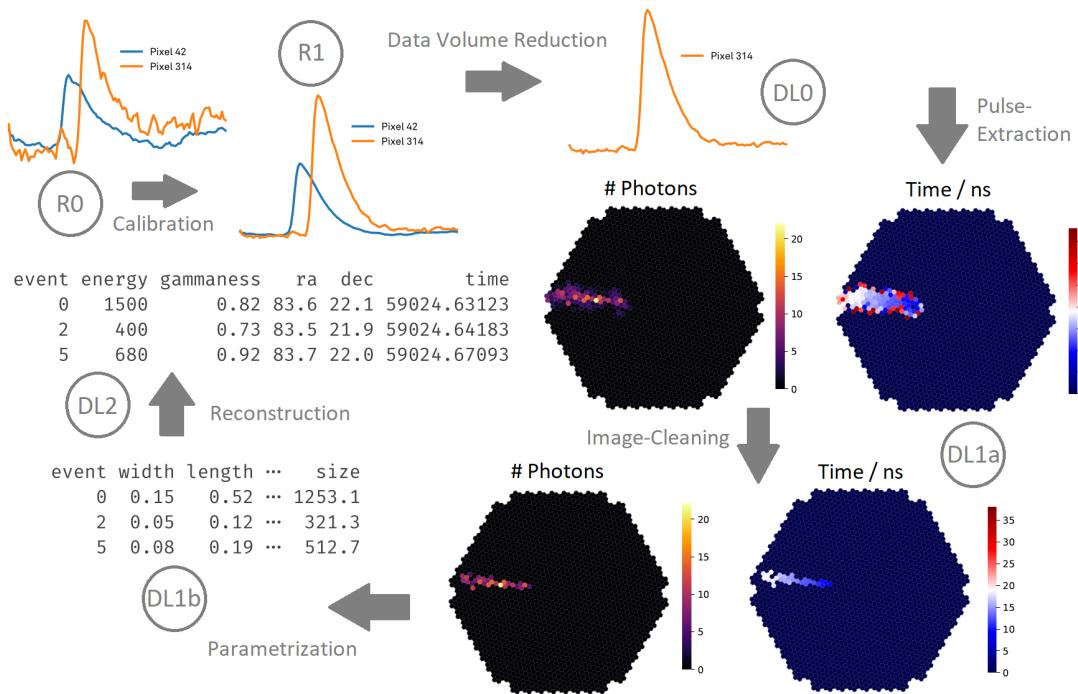


Figure 3.1: Data levels in ctapipe. Raw data levels are denoted by a **R** and calibrated data levels by a **D**. The raw data first gets calibrated (**R0** → **R1**) and then reduced in volume by selecting waveforms (**R1** → **DL0**). From there the images are extracted (**DL0** → **DL1a**) and cleaned with a cleaning algorithm. This allows for parametrization of the events (**DL1a** → **DL1b**). The parametrized events can then be reconstructed (**DL1b** → **DL2**) [13, 10].

citation

3.2 This work's data processing pipeline

In this work, the data processing pipeline is as follows: First, several simulation data runs are selected and processed with ctapipe. Then, the datasets are merged and serve as a basis for a re-run of the combined dataset with various parameter combinations described in chapter 4. Each resulting dataset can then be processed on an array- or telescope data level, resulting in **DL1a** image plots, angular resolution and effective area or efficiency as well as metrics for the performance. An in-detailed description of how this helps to compare the performance of the different cleaning algorithms can be found in section 4.2. Figure 3.2 shows a schematic overview of the data processing pipeline of this work.

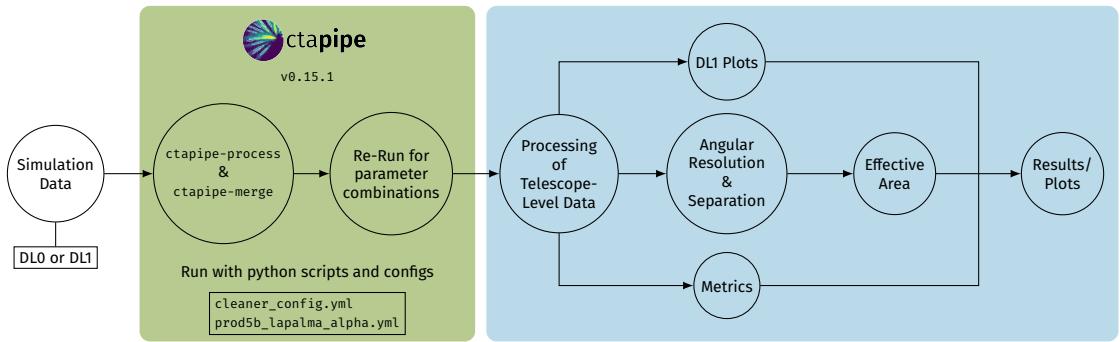


Figure 3.2: Schematic overview of the data pipeline used for this work. Single runs of the simulation data are processed with ctapipe-process and then merged via ctapipe-merge. The merged data is then processed on the array- or telescope data level resulting in scores for metrics as well as dl1 images and plots for the angular resolution and the effective area.

Finding Optimal Hyperparameters for the Cleaning Algorithms

4

This work aims to find optimal hyperparameters for the cleaning algorithms used in `ctapipe`. Optimizing the cleaning step (see section 3.1) can most certainly lead to a better reconstruction of the events in a dataset. Therefore a tweaking of the available parameters of each cleaner is necessary. In this chapter, I will first introduce the cleaning algorithms and their parameters in section 4.1 and then describe the procedure to find optimal hyperparameters in section 4.2.

4.1 Cleaning Algorithms

Version 0.15.1 of `ctapipe` features four cleaning algorithms, two of which are time-based. The `TailcutsImageCleaner` algorithm is the most basic algorithm of the four and serves as a good starting point for the development of new cleaning algorithms. Its first step is to select all pixels that are above a certain threshold, the `picture` or `core_threshold`. These pixels are the core part of the signal and are the brightest. The `TailcutsImageCleaner` algorithm then selects all pixels that are above the so-called `boundary_threshold` and are neighboring the core pixels. A visualization of the algorithm is shown in Figure 4.1 for the default values of the algorithm.

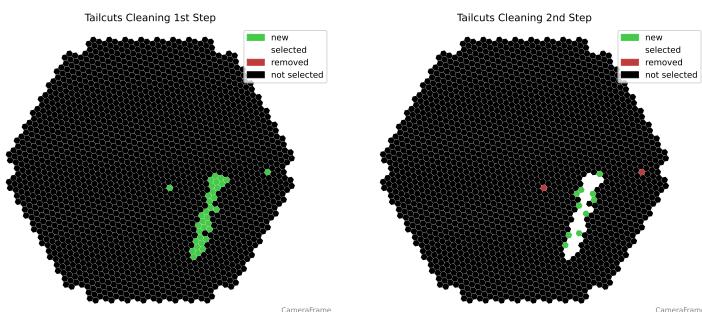


Figure 4.1: Visualization of the `TailcutsImageCleaner` algorithm for a MST Nectar-Cam image. First, all pixels above the `core_threshold` are selected. Then, all pixels neighboring the core pixels that are above the `boundary_threshold` are selected.

The `MARSImageCleaner` algorithm is very much based on the `TailcutsImageCleaner` algorithm and features an additional step, in that it also selects all neighbors of a neighbor of a core pixel, if they are above the `boundary_threshold`. The three steps for the `MARSImageCleaner` algorithm are shown in Figure 4.2 for the default values of the algorithm.

The `FACTImageCleaner` algorithm is a time-based cleaning algorithm that first selects all pixels that are above the `core_threshold`. Then, all pixels that have less than N neighbors are removed. The

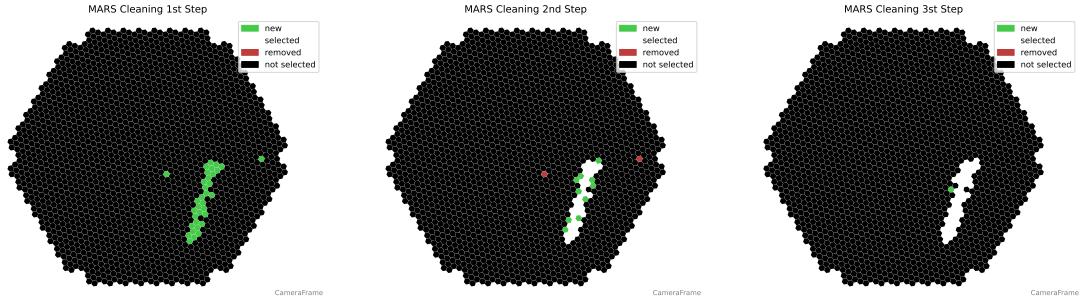


Figure 4.2: Visualization of the `MARSImageCleaner` algorithm for a MST NectarCam image.

`min_number_neighbors` parameter is set to 2 by default. For the third step, all pixels neighboring the remaining pixels that are above the `boundary_threshold` are selected. After that, all pixels that have less than N neighbors and that have arrived within a given timeframe are removed. The `time_limit` parameter is set to 5 ns by default. Again, all pixels with less than N neighbors are removed. The last step once again removes pixels with less than N neighbors, arriving within the given time limit. The visualization of the algorithm is shown in Figure 4.3 for the default values of the algorithm.

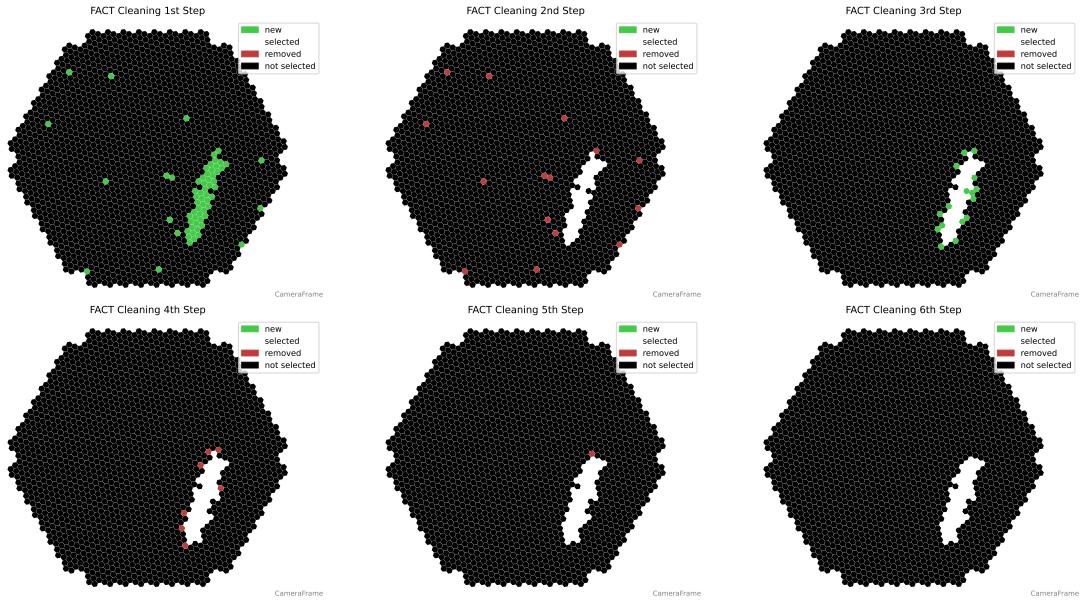


Figure 4.3: FACT cleaning steps

The `TimeConstrainedImageCleaner` algorithm is another time-based algorithm, coming from the MAGIC collaboration. It first selects all pixels that are above the `core_threshold`. Then, all pixels that have less than N neighbors are removed. The `min_number_neighbors` parameter is set to 1 by default. After that, all core pixels whose arrival times are within a given timeframe of the average arrival time. This `time_limit_core` parameter is set to 4.5 ns by default. As a fourth step, the `TimeConstrainedImageCleaner` algorithm finds all pixels above the `boundary_threshold`. Then, all pixels with less than N neighbors arriving within a given timeframe are removed. This `time_limit_boundary` parameter is set to 1.5 ns by

default. The visualization of the algorithm is shown in Figure 4.4 for the default values of the algorithm.

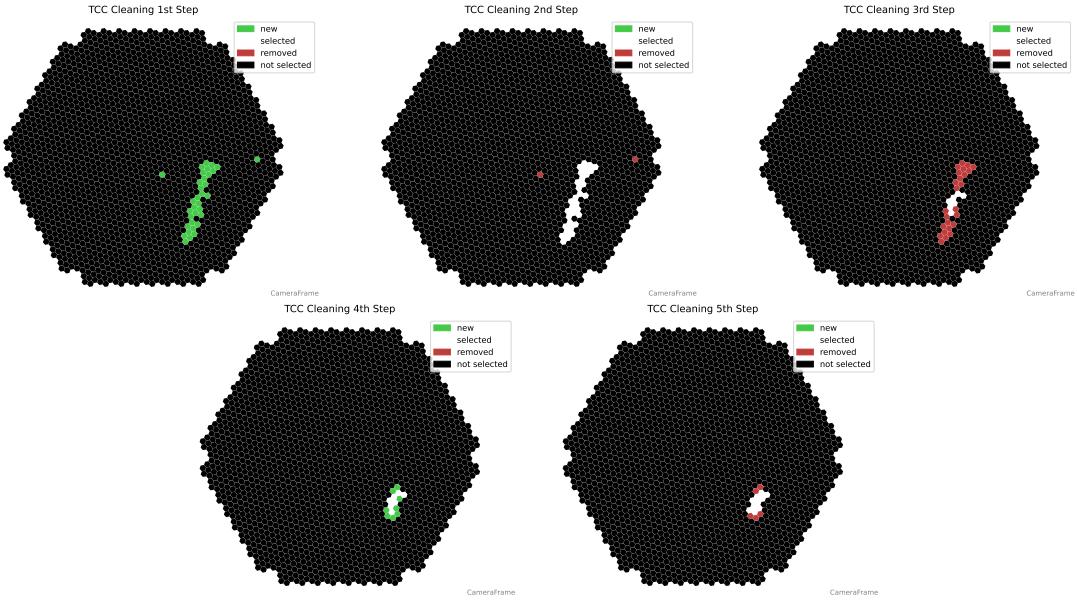


Figure 4.4: TCC cleaning steps

4.2 Hyperparameters

The hyperparameters of each cleaning algorithm are set in a specific config file. There, the user can set and change the parameters, shown in Table 4.1. TailcutsImageCleaner and MARSImageCleaner can be set-up with only three parameters: the `picture_threshold`, the `boundary_threshold` and `min_number_picture_neighbors`. The time-based algorithms have additional parameters: a `time_limit` for FACTImageCleaner and a `time_limit_core` as well as a `time_limit_boundary` for TimeConstrainedImageCleaner.

watch out for
text boundaries

To find the optimal hyperparameters I used the `ParameterGrid` class from the `sklearn.model_selection` module [14]. The `ParameterGrid` class is useful to create a dictionary of all possible combinations of a list of given parameters. These combinations are then written to a config file and processed with `ctapipe`. The output is hundreds of datasets equal to the number of combinations of the given parameters, each with a different setting for the cleaning performed on the data. Since it's impossible to find the optimal settings just by looking at the cleaned images of the datasets, a combined metric is necessary to evaluate each cleaner's performance for the parameter combinations.

In this work, I chose to first look at the efficiency $n_{\text{reco}}/n_{\text{total}}$, where n_{reco} is the number of reconstructed events and n_{total} the total number of events in the dataset. By taking the mean of the efficiency, one can sort the datasets by setting upper and lower bounds for the efficiency. The intervals lengths were chosen as 0.05, i. e. 5 %, resulting in a total of 20 intervals.

For each interval, the mean angular resolution is calculated. The dataset with the lowest mean angular resolution is the best performing for each interval. Not all events will ever be properly reconstructed for any parameter combination, so there won't be a mean angular resolution for all intervals, but only a

Table 4.1: The four cleaning algorithms and their hyperparameters. Being the most basic algorithms, TailcutsImageCleaner and MARSImageCleaner have only three parameters, while the FACTImageCleaner and TimeConstrainedImageCleaner algorithms have one and two additional time-based parameters, respectively. This table shows the default values, as they are implemented in the ctapipe source code for version 0.15.1.

Cleaning Algorithm	Hyperparameter	Default Values
TailcutsImageCleaner	picture_threshold	7 p.e.
	boundary_threshold	5 p.e.
	min_number_picture_neighbors	0
MARSImageCleaner	picture_threshold	7 p.e.
	boundary_threshold	5 p.e.
	min_number_picture_neighbors	0
FACTImageCleaner	picture_threshold	4 p.e.
	boundary_threshold	2 p.e.
	time_limit	5 ns
	min_number_picture_neighbors	2
TimeConstrainedImageCleaner	picture_threshold	7 p.e.
	boundary_threshold	5 p.e.
	time_limit_core	4.5 ns
	time_limit_boundary	1.5 ns
	min_number_picture_neighbors	1

subset. Also, the mean angular resolution may be higher when more events were reconstructed. As a result, a trade-off between efficiency and angular resolution is necessary.

Results

5

6

Conclusions and Outlook

Bibliography

1. B. P. Abbott et al. (LIGO Scientific Collaboration and Virgo Collaboration Collaboration). “Observation of Gravitational Waves from a Binary Black Hole Merger.” *Phys. Rev. Lett.* **116**, 6 2016, page 061102.
DOI: [10.1103/PhysRevLett.116.061102](https://doi.org/10.1103/PhysRevLett.116.061102). <https://link.aps.org/doi/10.1103/PhysRevLett.116.061102>
2. S. Abdollahi et al. “Fermi Large Area Telescope Fourth Source Catalog.” *The Astrophysical Journal Supplement Series* **247**:1, 2020, page 33.
DOI: [10.3847/1538-4365/ab6bcb](https://doi.org/10.3847/1538-4365/ab6bcb). ARXIV: [1902.10045 \[astro-ph.HE\]](https://arxiv.org/abs/1902.10045). <https://doi.org/10.3847/1538-4365/ab6bcb>
3. J. A. Aguilar, J. Yang, and S. Bravo. *Neutrinos and gamma rays, a partnership to explore the extreme universe*. IceCube/WIPAC. 2016.
<https://icecube.wisc.edu/news/view/455> visited on 2022-08-05
4. K. Brügge. “Unmasking the Gamma-Ray Sky. Comprehensive and Reproducible Analysis for Cherenkov Telescopes.” PhD thesis. Technische Universität Dortmund, 2019
5. *CTA Technology*. Cherenkov Telescope Array Consortium.
https://www.cta-observatory.org/wp-content/uploads/2019/12/CTA-Specifications_v08_formatted.pdf visited on 2022-08-10
6. *CTAO’s site in the northern hemisphere*. Cherenkov Telescope Array Consortium.
<https://www.cta-observatory.org/about/array-locations/la-palma/> visited on 2022-08-10
7. *CTAO’s site in the southern hemisphere*. Cherenkov Telescope Array Consortium.
<https://www.cta-observatory.org/about/array-locations/chile/> visited on 2022-08-10
8. C. Evoli. *The Cosmic-Ray Energy Spectrum*. 2020.
DOI: [10.5281/zenodo.4396125](https://doi.org/10.5281/zenodo.4396125). <https://doi.org/10.5281/zenodo.4396125>
9. S. Funk. “Ground- and Space-Based Gamma-Ray Astronomy.” *Annual Review of Nuclear and Particle Science* **65**:1, 2015, pages 245–277.
DOI: [10.1146/annurev-nucl-102014-022036](https://doi.org/10.1146/annurev-nucl-102014-022036). <https://doi.org/10.1146/annurev-nucl-102014-022036>
10. J. Hackfeld. “Analyzing the Data Volume Reduction for the LST-1 Prototype of the Cherenkov Telescope Array.” MA thesis. Ruhr-Universität Bochum, 2021
11. K. Kosack et al. *cta-observatory/ctapipe: v0.15.0 – 2022-05-23*. Version v0.15.0. 2022.
DOI: [10.5281/zenodo.6572720](https://doi.org/10.5281/zenodo.6572720). <https://doi.org/10.5281/zenodo.6572720>
12. M. S. Longair. *High energy astrophysics: an informal introduction for students of physics and astronomy*. Cambridge University Press, Cambridge [England] ; New York, 1981.
ISBN: 9780521235136 9780521280136

Bibliography

13. M. Nöthe. “Monitoring the High Energy Universe. Open, Reproducible, Machine Learning Based Analysis for the first G-APD Cherenkov Telescope.” PhD thesis. Technische Universität Dortmund, 2020
14. F. Pedregosa et al. “Scikit-learn: Machine Learning in Python.” *Journal of Machine Learning Research* 12, 2011, pages 2825–2830
15. *The technology behind the next generation very-high energy gamma-ray detector.* Cherenkov Telescope Array Consortium.
<https://www.cta-observatory.org/project/technology/> visited on 2022-08-11

Glossary

AGN active galactic nucleus. 2

CR cosmic rays. 1, 2

CTA Cherenkov Telescope Array. 4–8

EAS extensive air shower. 4, 5

ESO European Southern Observatory. 5

Fermi-LAT Fermi Large Area Telescope. 2

GRB gamma ray burst. 2

H. E. S. S. High Energy Stereoscopic System. 2

IACT Imaging Air Cherenkov Telescope. 2, 4, 5

LST Large-Sized Telescope. 6, 7

MAGIC Major Atmospheric Gamma-Ray Imaging Cherenkov. 2, 12

MST Medium-Sized Telescope. 6, 7, 11, 12

ORM Observatorio del Roque de los Muchachos. 5

PMT photo multiplier tube. 6

SiPM silicon photomultiplier. 6

SNR supernova remnant. 2, 3

SST Small-Sized Telescope. 6, 7

VERITAS Very Energetic Radiation Imaging Telescope Array System. 2

VHE gamma rays very-high-energy gamma rays. 3

Appendix

1 Configurations for ctapipe

The listings below show the contents of the default configuration files used for ctapipe with for each cleaning algorithm respectively.

TailcutsImageCleaner:

```
DataReader:  
    transform_image: true  
    transform_peak_time: true  
    write_images: false  
    write_parameters: true  
    write_raw_waveforms: false  
    write_showers: true  
  
ProcessorTool:  
    progress_bar: true  
  
CameraCalibrator:  
    image_extractor_type: NeighborPeakWindowSum  
  
ImageProcessor:  
    image_cleaner_type: TailcutsImageCleaner  
  
    TailcutsImageCleaner:  
        picture_threshold_pe:  
            - [type, "LST*", 8.5]  
            - [type, "MST*NectarCam", 9.0]  
        boundary_threshold_pe:  
            - [type, "LST*", 4.75]  
            - [type, "MST*NectarCam", 4.5]  
        keep_isolated_pixels: false  
        min_picture_neighbors: 2  
  
    ImageQualityQuery:  
        quality_criteria:  
            - ["enough_pixels", "np.count_nonzero(image) > 2"]  
            - ["enough_charge", "image.sum() > 50"]  
  
ShowerProcessor:  
    reconstructor_type: HillasReconstructor  
    HillasReconstructor:  
        StereoQualityQuery:  
            quality_criteria:  
                - [enough intensity, "parameters.hillas.intensity > 50"]  
                - [Positive width, "parameters.hillas.width.value > 0"]
```

Appendix

```
- [enough pixels, "parameters.morphology.num_pixels > 3"]
- [not clipped, "parameters.leakage.intensity_width_2 < 0.5"]
```

MARSImageCleaner:

```
DataReader:
    transform_image: true
    transform_peak_time: true
    write_images: false
    write_parameters: true
    write_raw_waveforms: false
    write_showers: true

ProcessorTool:
    progress_bar: true

CameraCalibrator:
    image_extractor_type: NeighborPeakWindowSum

ImageProcessor:
    image_cleaner_type: MARSImageCleaner

MARSImageCleaner:
    picture_threshold_pe:
        - [type, "LST*", 8.5]
        - [type, "MST*NectarCam", 9.0]
    boundary_threshold_pe:
        - [type, "LST*", 4.75]
        - [type, "MST*NectarCam", 4.5]
    keep_isolated_pixels: false
    min_picture_neighbors: 2

ImageQualityQuery:
    quality_criteria:
        - ["enough_pixels", "np.count_nonzero(image) > 2"]
        - ["enough_charge", "image.sum() > 50"]

ShowerProcessor:
    reconstructor_type: HillasReconstructor
    HillasReconstructor:
        StereoQualityQuery:
            quality_criteria:
                - [enough intensity, "parameters.hillas.intensity > 50"]
                - [Positive width, "parameters.hillas.width.value > 0"]
                - [enough pixels, "parameters.morphology.num_pixels > 3"]
                - [not clipped, "parameters.leakage.intensity_width_2 < 0.5"]
```

FACTImageCleaner:

```
DataReader:
    transform_image: true
    transform_peak_time: true
    write_images: false
    write_parameters: true
    write_raw_waveforms: false
    write_showers: true
```

```

ProcessorTool:
  progress_bar: true

CameraCalibrator:
  image_extractor_type: NeighborPeakWindowSum

ImageProcessor:
  image_cleaner_type: FACTImageCleaner

FACTImageCleaner:
  picture_threshold_pe:
    - [type, "LST*", 8.5]
    - [type, "MST*NectarCam", 9.0]
  boundary_threshold_pe:
    - [type, "LST*", 4.75]
    - [type, "MST*NectarCam", 4.5]
  keep_isolated_pixels: false
  min_picture_neighbors: 2

ImageQualityQuery:
  quality_criteria:
    - ["enough_pixels", "np.count_nonzero(image) > 2"]
    - ["enough_charge", "image.sum() > 50"]

ShowerProcessor:
  reconstructor_type: HillasReconstructor
  HillasReconstructor:
    StereoQualityQuery:
      quality_criteria:
        - [enough intensity, "parameters.hillas.intensity > 50"]
        - [Positive width, "parameters.hillas.width.value > 0"]
        - [enough pixels, "parameters.morphology.num_pixels > 3"]
        - [not clipped, "parameters.leakage.intensity_width_2 < 0.5"]

```

TimeConstrainedImageCleaner:

```

DataWriter:
  transform_image: true
  transform_peak_time: true
  write_images: false
  write_parameters: true
  write_raw_waveforms: false
  write_showers: true

ProcessorTool:
  progress_bar: true

CameraCalibrator:
  image_extractor_type: NeighborPeakWindowSum

ImageProcessor:
  image_cleaner_type: TimeConstrainedImageCleaner

TimeConstrainedImageCleaner:
  picture_threshold_pe:
    - [type, "LST*", 8.5]
    - [type, "MST*NectarCam", 9.0]

```

Appendix

```
boundary_threshold_pe:
  - [type, "LST*", 4.75]
  - [type, "MST*NectarCam", 4.5]
keep_isolated_pixels: false
min_picture_neighbors: 2

ImageQualityQuery:
  quality_criteria:
    - ["enough_pixels", "np.count_nonzero(image) > 2"]
    - ["enough_charge", "image.sum() > 50"]

ShowerProcessor:
  reconstructor_type: HillasReconstructor
HillasReconstructor:
  StereoQualityQuery:
    quality_criteria:
      - [enough intensity, "parameters.hillas.intensity > 50"]
      - [Positive width, "parameters.hillas.width.value > 0"]
      - [enough pixels, "parameters.morphology.num_pixels > 3"]
      - [not clipped, "parameters.leakage.intensity_width_2 < 0.5"]
```

The following listing shows the contents of the `prod5b_lapalma_alpha.yml` configuration file, which is used to set the allowed telescope IDs for `ctapipe-process`.

```
# telescope ids of the 4 LST + 9 LST La Palma alpha configuration in Prod5b
# (out of 84 telescopes total)
EventSource:
  allowed_tels: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 19, 35]
```

Acknowledgements

Eidesstattliche Versicherung

(Affidavit)

Name, Vorname
(surname, first name)

Bachelorarbeit
(Bachelor's thesis)

Titel
(Title)

Matrikelnummer
(student ID number)

Masterarbeit
(Master's thesis)

Ich versichere hiermit an Eides statt, dass ich die vorliegende Abschlussarbeit mit dem oben genannten Titel selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

I declare in lieu of oath that I have completed the present thesis with the above-mentioned title independently and without any unauthorized assistance. I have not used any other sources or aids than the ones listed and have documented quotations and paraphrases as such. The thesis in its current or similar version has not been submitted to an auditing institution before.

Ort, Datum
(place, date)

Unterschrift
(signature)

Belehrung:

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -).

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird ggf. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen:

Official notification:

Any person who intentionally breaches any regulation of university examination regulations relating to deception in examination performance is acting improperly. This offense can be punished with a fine of up to EUR 50,000.00. The competent administrative authority for the pursuit and prosecution of offenses of this type is the Chancellor of TU Dortmund University. In the case of multiple or other serious attempts at deception, the examinee can also be unenrolled, Section 63 (5) North Rhine-Westphalia Higher Education Act (*Hochschulgesetz, HG*).

The submission of a false affidavit will be punished with a prison sentence of up to three years or a fine.

As may be necessary, TU Dortmund University will make use of electronic plagiarism-prevention tools (e.g. the "turnitin" service) in order to monitor violations during the examination procedures.

I have taken note of the above official notification.*

Ort, Datum
(place, date)

Unterschrift
(signature)

***Please be aware that solely the German version of the affidavit ("Eidesstattliche Versicherung") for the Bachelor's/ Master's thesis is the official and legally binding version.**