Diploma Thesis

Camera-specific Image Denoising

Michael Schober

Eberhard Karls Universität Tübingen

Tübingen, October 2013

Prof. Dr. Andreas Schilling
Graphische-Interaktive Systeme
Wilhelm Schickard Institut für Informatik
Eberhard Karls Universität Tübingen

Prof. Dr. Bernhard Schölkopf
Abteilung Empirische Inferenz
MPI für Intelligente Systeme Tübingen

WS GR\S Wilhelm Schickard Institut für Informatik
GRaphisch-Interaktive Systeme

MAX-PLANCK-GESELLSCHAFT

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich, Michael Schober, die vorliegende Arbeit selbstständig angefertigt, keine anderen als die angegebenen Hilfsmittel benutzt und alle Stellen, die dem Wortlaut oder Sinne nach anderen Werken entnommen sind, durch Angabe der Quellen als Entlehnung kenntlich gemacht habe. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

I was just guessing
At numbers and figures
Pulling the puzzles apart

Coldplay. „The Scientist.“ *A Rush of Blood to the Head.* Capitol, 2002.

I've done the math enough to know the dangers of our second guessing
Doomed to crumble unless we grow, and strengthen our communication

Tool. „Schism.“ *Lateralus.* Volcano, 2001.

Abstract

Images captured with digital cameras are corrupted by noise due to the properties of the physical measurement process. Traditional image denoising algorithms have been based either solely on image statistics or solely on noise statistics, but not both at the same time. Secondly, the connections between general purpose image denoising algorithms and camera-specific denoising algorithms have not been investigated. In this work, image denoising with realistic camera noise will be examined, specifically for application in astro-photography. It will be shown that high-quality images can be obtained through careful calibration and residual noise is significantly less than most commonly assumed. Additionally, a new machine learning based approach to camera-specific image denoising will be presented.

Zusammenfassung

Mit Digitalkameras aufgenommene Bilder enthalten zwangsläufig Signalrauschen. Traditionellerweise haben Algorithmen zum Entrauschen entweder Bildstatistiken oder Informationen der Rauschquelle in ihre Modelle inkorporiert, aber bisher wurden noch nicht beide Informationsquellen gleichzeitig verwendet. Des weiteren gibt es bisher keine Untersuchung der Zusammenhänge allgemeiner und kamera-spezifischer Entrauschungs-Algorithmen. In dieser Arbeit wird das Entrauschen von Bildern mit realistischem Kamerarauschen untersucht, speziell für die Anwendung im Bereich der Astrofotografie. Es wird gezeigt, dass hochqualitative Bilder durch sorgfältige Kalibrierung aufgenommen werden können und dass das Residuenrauschen deutlich geringer ist, als es bei den meisten Arbeiten angenommen wird. Weiterhin wird ein neuer, auf maschinellem Lernen beruhender Algorithmus zum kameraspefizischen Entrauschen vorgestellt.

Acknowledgments

First of all, I thank Prof. Dr. Bernhard Schölkopf for giving me the opportunity to write my diploma thesis at the Max Planck Institute for Intelligent Systems and my colleagues and friends here for many hours of insightful and valuable discussions. It is fair to say that during my time here I have learned almost as much as during all of my studies before and for this I will forever be grateful. Secondly, I want to thank him for suggesting this topic and supervising this thesis, and even more so for inviting me to meet him during his holiday in Ibach and for helping me taking my very first images of the Milky Way. I was able to catch a glimpse into an unknown world to me and also catch some of the excitement that I have not known before.

My warmest gratitude also go out to Prof. Dr. Andreas Schilling, who was willing to supervise this thesis for the University of Tübingen, for his trust that something interesting will come out of this. I also want to thank him for his lecture on Machine Learning in the spring semester of 2012 that I was happy to take part in, which I very much enjoyed, and which encouraged me to keep delving into machine learning.

I also want to thank Prof. em. Dr. Hanns Ruder as well as Dr. Michael Hirsch, which helped me with specific questions about astronomy and astro-photograpy in particular. Of my other colleagues, I have and want to mention Nicole Schmeißer, Edgar Klenske, Dr. Philipp Hennig, Christopher Burger and Christian Schuler. They have spent a lot of time listening to my problems, giving insightful comments or just general help whenever I needed it. I specially want to thank Christopher Burger who helped me even after he received his doctoral degree and funded his own company — something which is arguably more important to focus on than on a diploma thesis. And, of course, Christian Schuler, who spent many hours with me looking at source code, experiment setups, and results, and helped me making sense when the numbers just did not want to add up at all. For all this support I am deeply grateful.

Finally, I want to thank my family and friends for putting up with me over the years and specially during this last stressful phase when I often had to cancel plans due to long hours on the project. I know that they are there whenever I need them and this is the greatest gift of all. That, and our nights out, of course. Thank you.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Problem Description

Digital cameras are ubiquitous in modern daily life. Professional photographers have long been switching to digital single-lens reflex cameras. Digital compact cameras have replaced traditional analogue film cameras in stores. Almost all mobile phones available for purchase come with a digital camera included.

And since all digital cameras are physical measurement instruments, the recorded images necessarily suffer from measurement noise.

In the classical problem formulation of image denoising, noise is described as the deviation of the recorded signal from the actual signal present during the time of recording. In its simplest form, the problem can be described with one formula:

$$y = x + n \tag{1.1}$$

In this equation, x represents the noise-free image signal, n is some sort of pixel-wise additive noise and y is the recorded image at hand. Naively, this can be viewed as an under-determined system of linear equations, since the two unknown variables x and n have to be inferred from only one given variable y.

In the past, many algorithms have been proposed to estimate the original image x from a noisy observation y. Many of these algorithms work under very general assumptions and, consequently, cannot take into account properties of the recording process.

The goal of this thesis is to investigate how the situation changes if not a general noise model is assumed, but it is known that the noise is introduced by a specific camera. In this case, it is expected that existing methods can be improved upon. Secondly, it is necessary to identify how the case of camera-specific image denoising is related to the general approach and what implications this has for general denoising methods.

To this end, a specific problem is presented and analyzed: the case of image denoising in astro-photography.

1.2 The Importance of Camera-specific Image Denoising

To understand the importance of camera-specific image denoising, it is key to ask who the users of image denoising are. In many cases, these are the photographers of the images themselves. This is enlightening, because then one can assume that most users of image denoising algorithms have access to more information about the noise in the input data than the general model implies.

Generally speaking, there are conceptually two different sources of noise in a digital image. The first source of noise stems from the property of light itself: the number of photons emitted from an object per time period is not deterministic, but in fact follows a Poisson distribution with parameter λ proportional to its brightness.

Secondly, there is noise introduced by the measurement instrument, i.e., the digital camera. Every digital camera suffers from a variety of noise sources due to manufacturing imperfections that will always be introduced into the measurement.

When the light source is bright enough — in other words, when λ is big enough — the Poisson distribution can be approximated with a Gaussian distribution of mean and variance λ. Together with the noise term, this can be treated as the sum of two Gaussians, which can in turn be treated as a constant term for the mean value and a residual term for the noise.

Without other prior knowledge, this noise described in the last paragraph may be assumed to be *additive white Gaussian noise*, or *AWGN* for short. (AWGN will be described in Section 2.1.) However, in the following it will be argued that camera noise is not — and cannot possibly be — additive white Gaussian noise.

For instance, noise generated by thermal energy is necessarily positive: only additional electrons can be freed and will consequently be introduced into the signal. (The image capturing process will be discussed in Sections 3.1 and 3.2.) This obviously violates the zero-mean assumption of the AWGN model. Additionally, it is unlikely that all pixels have the same standard deviation of noisiness, which in turn violates the whiteness assumption. In fact, so-called *hot pixels* and *dead pixels* are a well-known camera defects among photographers and these pixels can be thought of as impulse noise at known locations. At the very least, this additional information can help improve denoising performance.

On the other hand, it is also unlikely that the noise characteristics will be shared even among cameras of the same model type. This is due to manufacturing inaccuracies, which are very hard to reduce much further on the level of precision needed or the purity of materials involved.

And finally, it is also likely that the properties of even the same camera will change over time due to the natural aging process of the material or its operation handling. Dead pixels, for

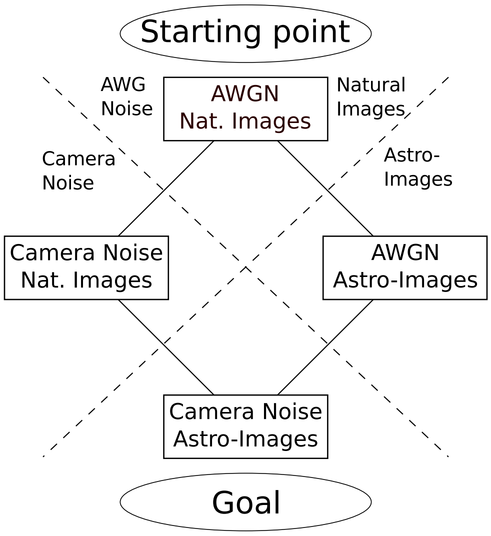


Figure 1.1: Possible configurations of image and noise statistics

instance, get stuck at a certain point in time, which the photographer will try to keep track of and take action accordingly, e.g., by interpolating pixel values.

However, with access to the camera, users can try to measure the introduced noise and, therefore, calibrate the images accordingly with this additional information. The AWGN assumption is only a fall-back assumption, which should be refined as more data becomes available as is the case in this scenario.

1.3 Camera-specific Image Denoising in Astro-Photography

Similar to the question of who the users of denoising algorithms are, is the question when denoising algorithms are applied, put differently: when is noise most problematic in digital images?

It will be argued in Section 4.1 that this is the case for astronomical imaging. There, the signal is very faint, with only a few photons emitted during a long exposure. At the same time, noise accumulates during exposure time which will deteriorate the signal further, but which will also make the noise pattern of a given camera more distinct.

Therefore, the general approach given in Equation (1.1) can be specialized to incorporate both additional knowledge from the camera noise distribution as well as knowledge of the special distribution of the input signal. Figure 1.1 illustrates this process.

1.4 Outline

Chapter 2 discusses background and related work of the general image denoising problem. Mechanics and implications thereof for image denoising of digital cameras are presented in Chapter 3. In Chapter 4, properties of natural and astronomical images are discussed. It will be argued that camera-specific image denoising is most important in astro-photography and further consequences for the denoising method in this thesis are detailed. Next, the precise models and methods used in this thesis are provided in Chapter 5. This chapter describes all the theoretical aspects of the experimental evaluations, which are in turn presented in Chapter 6. Chapter 7 summarizes the results of this thesis and ends in conclusions of this research.

2 Image Denoising

2.1 Principles of Image Denoising

As argued in Chapter 1, digital images are necessarily corrupted by noise, i.e., there are mismatches in reported pixel values that do not correspond to the radiance that was inherent in the scene during the time of signal recording. Depending on the exact process of image acquisition, this noise might have very different characteristics and properties.

As has been stated in Section 1.2, a very common and reasonable assumption in image denoising is a pixel-wise additive error drawn from a Gaussian distribution with mean value zero and some standard deviation σ, that is, for each pixel i the reported pixel value y_i depends on the true signal x_i through

$$n_i \sim \mathcal{N}(0, \sigma) \tag{2.1}$$

$$y_i = x_i + n_i \tag{2.2}$$

Furthermore, the n_i are assumed to be independent for each pixel. This type of noise characterization is shared among many signal processing systems and is commonly known as *additive white Gaussian noise (AWGN).* Often, σ is assumed to be known.

Image denoising is the task to find or construct good estimators f, which, given the noisy observation $\tilde{I} = \{y_i \,|\, i = 1, \ldots, N\}$, produce an estimate $f(\tilde{I}) = \hat{I} = \{\hat{y}_i \,|\, i = 1, \ldots, N\}$ that is *less noisy* in some desirable way. To this end, several image quality metrics have been proposed over the years. However, the most common of these metrics is based on the *mean squared error (MSE)* between estimated image and true signal. It is common in the image denoising community to use an inverted and logarithmic scale called the *Peak Signal-to-Noise Ratio (PSNR)* , given via

$$\mathrm{PSNR}(I, \hat{I}) = 10 \cdot \log_{10} \left(\frac{I_{\mathrm{MAX}}^2}{\frac{1}{N} \sum_{i=1}^{N} (I_i - \hat{I}_i)^2} \right) \mathrm{dB} \tag{2.3}$$

where I_{MAX} denotes the maximal possible value of any pixel i (e.g. $2^8 - 1 = 255$ for 8-bit JPEG images).

Given that the evaluation metric depends monotonically on the MSE, it is clear that any denoising attempt has to consider the context of the signal, i.e., not just a single pixel but some sort of collection of pixels. To see this, recall that the maximum likelihood estimator is an UMVU-estimator (uniformly minimum-variance unbiased estimator). If there is only one data point, the maximum likelihood under zero-mean Gaussian noise is just the observation itself. Therefore, if there is not a sequence of images taken under the same condition, one cannot remove any noise based solely on a single pixel.

Consequently, it is necessary to consider additional information, such as the context of the pixel or other insight of the noise source. Inspecting (2.2), it is clear that there are two possible information sources in principle:

1. Noise statistics
2. Image statistics

In the case of AWGN, the first option is actually not feasible, because if the noise is not correlated, this method falls back to pixel-wise information, which has already been argued to be infeasible. However, for many applications, AWGN is a good basic assumption, because combining several noise sources will likely lead to some kind of pixel-wise Gaussian noise due to the central limit theorem. Therefore, one necessarily has to apply information from the image content.

2.2 Taxonomy of Image Denoising Methods

The following is a possible taxonomy of various methods for image denoising, each of which will be discussed shortly in the following:

1. Use hand-crafted methods based on general arguments about signals, i.e., use general filtering techniques
2. Use global statistics of images
3. Use only image internal statistics
4. Use global statistics of images, but restricted conditional on the image internal statistics

As depicted in Figure 2.1, one can think of these methods as being refinements to more and more specialized cases. In this sense, it is expected that denoising performance will improve when the more constrained assumptions do in fact hold. At the same time, there is also a potential

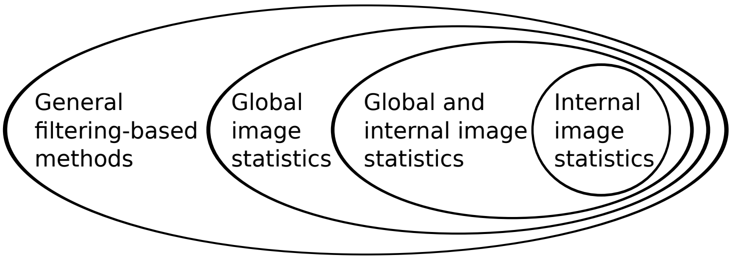

General filtering-based methods
Global image statistics
Global and internal image statistics
Internal image statistics

Figure 2.1: A possible taxonomy of various methods for image denoising. Methods belonging to inner sets can be thought of as specializations of more general methods.

problem of making too strong assumptions and therefore constraining the solution too much, leading to a decreased performance in the end. In a very broad sense, this can be thought of as a *variance* versus *bias* trade-off, which is a common problem in statistical methods. However, a popular algorithm, that achieved *state-of-the-art* results for a long time, mostly relies on internal statistics of images [1]. So it seems that the danger of biasing the estimation is not very big.

In the following, each type of method is discussed.

2.2.1 Filtering

Filtering based methods rely on general arguments which either hold for all kinds of digital images or in some cases even for digital signals in general. For example, a very basic assumption that is likely to hold for many cases is *smoothness*, i.e., pixel values will not change drastically from pixel to pixel. In this case, neighboring pixels can be thought of as similar samples to the pixel that is currently denoised.

Algorithms in this category include:

- Mean filtering
- Median filtering
- Bilateral filtering [2]
- Anisotropic diffusion [3]

The upsides of these methods are their speed and generality. Furthermore, no preprocessing of any kind needs to be applied. However, due to their generality, their results are often inferior to task-specific algorithms.

2.2.2 Algorithms applying Global Image Statistics

Methods relying on global image statistics also make general assumptions about images. However, in these types of algorithms, there is at least one part of the computation that has necessarily been applied to clean sample images to extract general statistics of digital images. For instance, there could be the need for a dictionary of clean image patches or a likelihood function for pixel neighborhoods based on Markov Random Fields. Additionally, algorithms that exploit knowledge of natural image statistics (e.g., sparsity in Fourier space or according to some wavelet basis) which have been found empirically will also be considered part of this category, since empirical studies of natural images have to be performed prior to formulating this denoising approaches.

Methods in this area include:

- Decomposition-based methods, such as wavelet or dictionary-based methods [4, 5]
- Learned-filter based methods [6, 7]

It is observable that these methods usually perform better than general filtering based approaches, since the outputs are pushed towards realistic image models. Additionally, for many instances of algorithms in this category, computation time stays low, because the global image statistics need only be computed once.

2.2.3 Algorithms applying Internal Statistics

Taking this kind of reasoning to the extreme leads to methods that try to rely only or mostly on statistics found in the image itself. Algorithms in this class include Non-local Means [8] and the state-of-the-art algorithm BM3D [1]. Both algorithms operate on a per-patch basis which will be described in more detail in Section 5.2.

Another common feature is the dependency on *self-similarity* of natural images. In some sense, it could also be argued that this is a very general signal processing idea. Additionally, BM3D in particular also applies wavelet coefficient shrinking to the image patches, which is an idea and observation that is based on general statistics of natural images. However, since only image patches from within the noisy image itself are considered, these methods will be considered to be working on internal statistics only.

These methods have been shown to work among the best in practice and have the additional advantage that no extra image statistics are required during runtime. However, these methods tend to have long run times, since a lot of local evaluations in input images have to be performed.

2.2.4 Combining External and Internal Image Statistics

It is straightforward to imagine that an algorithm combining external and internal image statistics might be able to have the best of two worlds and therefore outperform both algorithms that solely rely on global or local image statistics. However, as has been shown by [9], a very large number of similar patches are expected to be required to achieve comparable results. On the other hand, good results have been reported in the related image restoration task of super-resolution [10].

The approach, which is introduced and discussed in this thesis, will also belong to this category of denoising algorithms.

2.3 Image Denoising for other Types of Noise

Even though the case of additive white Gaussian noise is the type most considered in literature, there are also other types of noise models for images, which have been studied. These types of noise usually occur under more specific conditions and, therefore, special algorithms can be applied to incorporate the additional information that is given through the more constrained noise type.

In the following, two types of noise are presented which are of special interest in the setting of camera-specific noise removal and astronomical imaging.

2.3.1 Mixed Poisson-Gaussian Noise

The number of photons emitting from a light source is not deterministic, but is in fact distributed according to a *Poisson*-distribution with some parameter λ that is proportional to the intensity of the source. The Poisson distribution is given by

$$p(x; \lambda) = \frac{\lambda^x}{x!} e^{-\lambda} \tag{2.4}$$

Since another expression for signal-to-noise ratio is the expected value of the signal divided by its standard deviation, one can see that the signal-to-noise ratio of a Poisson-distributed process will increase with $\sqrt{\lambda}$ as λ increases. Therefore, in a well-lit environment, the noise introduced by the random arrival of photons can be neglected and the AWGN dominates the signal deterioration. However, in a setting, where there is only little light available, such as astronomical imaging, a significant amount of noise will also be introduced by the stochastic nature of the signal itself.

Therefore, in low light environments, a mixed Poisson-Gaussian noise model is assumed which is defined as:

$$x_i \sim \mathcal{P}(\lambda_i) \tag{2.5}$$

$$n_i \sim \mathcal{N}(0, \sigma) \tag{2.6}$$

$$y_i = \alpha x_i + n_i \tag{2.7}$$

where α is a scaling parameter.

A number of algorithms have been proposed in recent years to specifically deal with this kind of image noise [11, 12]. Other approaches apply suitable transformations to reduce the problem to the case of regular AWG noise [13].

2.3.2 Impulse Noise

Impulse noise models come in various forms, but they all share a fundamentally different assumption of degradation as compared to general additive noise. In this setting, a certain percentage p of pixels is assumed to be completely void of any signal. This could be due to the transmission pipeline, or due to complete hardware failures in a capturing device. The corrupted pixels could either be fixed to a specific value, Bernoulli-distributed (*Salt-and-Pepper noise*) or Gaussian distributed (*Impulse noise*). Additionally, in some models even the pixels with signal have an additional noise term as in (2.2).

Since the properties of this kind of noise are quite different to additive noise models, a whole different set of algorithms have been proposed in the literature for this denoising problem [14, 15, 16, 17]. Most importantly, any type of mean filter is not expected to work for this noise model, since the locations of noisy pixels are not known and cannot be excluded from the mean. Instead, algorithms focus on variations of median filtering, many with an additional aspect of corrupt pixel and edge detection.

For actual camera systems, very noisy pixels will essentially be Bernoulli distributed, since the minimum and maximum signal is limited, thereby representing some kind of Salt-and-Pepper noise.

2.4 Learning-based Denoising

Although it is clear, that certain assumptions and prior signal information must be used to be able to denoise an input image, there are different ways to express and incorporate this prior knowledge into the denoising algorithm. Secondly, if prior information is used in form of explicit statistical image models, there is a multitude of methods and formulations that have been developed by both, the statistics and machine learning community as well as the image processing community to represent this information.

Prior information can have different meanings in the following sense: it can refer to general assumptions, such as self-similarity or sparsity of the wavelet coefficients. At the same time, it can also refer to probability distributions over pixel-value combinations in actual images or small image patches. In the latter case, this probability distribution is referred to as *prior distribution*, or simply *prior*, over images or image patches, since it can be used for a multitude of image restoration tasks, given an additional likelihood and applying Bayes' rule of probability theory. For instance, in image denoising with AWGN, the likelihood is simply given by a multivariate Gaussian.

However, finding a good mathematical description of an image prior is a difficult research question and many different models have been proposed in the literature [6, 18, 19]. Additionally, priors can also be given indirectly or not necessarily in the form of a probability formulation. For instance, the former is the case in a neural network formulation of the image denoising problem [20, 7], whereas the latter can be found in a learned sparse dictionary of image patches as in [5].

As is certainly the case for the statistics of image patches, machine learning algorithms can be used to automatically extract important statistics of complex and large data. Many algorithms share the advantage that very little assumptions have to be made beforehand, thus limiting the danger of constraining the set of solutions too heavily, such that a good approximation might not be found in the latter case. Additionally, many algorithms allow to optimize the model to specific means, which is often given as a *loss function* in machine learning. Therefore, the same model structure can be used to optimally solve different tasks [21, 22].

For these reasons, this work will focus on machine learning algorithms to deal with complex statistics of images and camera noise.

3 Camera Noise

3.1 Modern Camera Technology

This section will give a brief schematic overview of contemporary digital camera technologies and possible implications for the expected noise characteristics of these imaging systems.

Currently, there are two kinds of general architectures for digital cameras:

1. *Charge-coupled devices (CCD)*

2. *Complementary metal oxide semiconductors (CMOS)* with *Active Pixel Sensors (APS)*

In the second case, CMOS refers more to the manufacturing process than the imaging process, whereas CCD and APS can both be thought of as the underlying signal capturing strategy.

For with both technologies, the initial capturing process is achieved in the same way. Arriving photons free electrons in the sensor element of each pixel due to the photoelectric effect. Freed electrons are collected during the integration time, i.e., during the exposure, and then somehow converted into a voltage, when, finally, an analog-to-digital converter (ADC) reports the measured intensity to the digital part of the imaging pipeline. The main difference between the two different architectures lies in the process of converting the electrons into a voltage.

In charge-coupled devices, pixels are processed in a two-step process, where first the charge of the bottom row is transfered to a charge amplifier which is then processed sequentially for all columns. Secondly, for each column the charge is shifted one row towards the bottom end, thus eventually converting all pixels one row at a time. A common analogy found in literature is the image of a system of conveyor belts, passing around buckets of water [23]. More technical and detailed information about charge-coupled devices can be found in, e.g., [24, 25, 26].

In contrast, in CMOS imaging sensors each pixel element already integrates some of the functionality in the pixel itself, hence the name active pixel sensor. The manufacturing process of CMOS imaging sensors resembles those of CPUs and thus a wide range of integrated circuits can be put onto the chip with little production overhead. At the time of their invention, this benefit was overshadowed by their poor performance — CMOS camera systems introduced a lot of noise to the signal. However, modern architectures improved both on general layout as well as production precision, such that new CMOS imaging chips can compete with the quality of classic CCD digital cameras. Moreover, there are additional advantages due to faster read-out times and the possibility to integrate other functionality directly into the hardware, such as white-balancing or even basic digital denoising strategies. Additional insights into the CMOS technology for image capturing is given in, e.g., [27, 28, 29, 30, 31].

These differences in technology should also hint at some possible differences for image denoising models: since the signal in charge-coupled devices is transfered over regions of the sensor , it is much more likely that part of the noise is correlated in such systems. And, in fact, this is often observed in actual cameras, where, for instance, a column gives poor performance starting from a certain location. This kind of defect is called *column defect*.

In CMOS based cameras, however, it is more likely that the noise distribution is not identical over the chip. This has indeed been observed very significantly specially in early cameras, where the noise introduced a clearly visible pattern on the recorded image [27].

3.2 Sources and Types of Noise in Digital Cameras

Although the underlying technology is different for CCD and CMOS imagers, most sources of noise occur in both types of devices and can therefore be introduced in a general manner.

Categorically, we can distinguish 2 different types of noise sources: *temporal noise* and *spatial noise*. Temporal noise sources are due to stochastic processes during exposure time. To this category belong:

Photon Shot Noise (PSN) As already mentioned in Section 2.3.1, photons are emitted from a light source not uniformly, but according to a Poisson distribution with parameter $\lambda \sim I$ proportional to the irradiance of the scene. Therefore, although the number of photons arriving at a given pixel will be correct in expectation, it will typically deviate from that value with a standard deviation in the order of $\sqrt{\lambda}$. Therefore, the SNR will be higher for scenes with high irradiance, or more accurately, it will improve with the square root of the irradiance, although the absolute value of the induced uncertainty will be higher.

Dark Current Shot Noise (DCSN) Due to thermal energy, some electrons will be freed without the arrival of corresponding photons. These are called *dark current*. This effect becomes dependent on exposure time (the longer the more dark current) and the sensor's temperature (the higher the more). This effect gets even more accentuated when working with a non-cooled camera, since the sensor will heat up during exposure and thus the amount of dark current per time interval will increase during exposure. This problem is often reduced in scientific CCD cameras, which incorporate a cooling mechanism that can control the temperature of the chip up to a high precision. Some retailers also offer cooling systems for some CMOS cameras, but this has not found wide-spread use yet.

Readout noise The transformation from the number of freed electrons to the final raw value which is reported by the pixel incorporates several steps, in each of which additional noise will be added by the electronic elements. However, these sources of noise are mutually independent, additive, and, most importantly, not so much depending on exposure time or sensor temperature, and can therefore be composedly treated as Gaussian noise. However, the exact inaccuracy added in each capture will still be not identical. Therefore, this noise source must still be regarded as a temporal one.

Conversely, *spatial noise sources* are non-uniformities introduced through pixel-wise differences on the sensor due to manufacturing inaccuracies and material irregularities. These noise sources could still be observed after capturing very carefully evenly illuminated scenes for many times and averaging the results. Of course, some non-uniformities are due to circumstances outside of the image sensor, e.g. some kinds of vignetting and the $\cos^4$ law. However, even if these causes would be computationally be accounted for, the sensor would still introduce non-uniformities of its own. To these noise sources belong:

Photo-Response Non-Uniformity (PRNU) Generally, the *raw value* or *digital number (DN)* that gets reported by the image sensor per pixel is proportional to the number of photons arriving on that pixel during exposure. These proportion can be described by the *gain factor*. In CCD cameras, this gain is usually fixed, whereas in CMOS cameras this can be adjusted by setting the so-called *ISO value*. However, there exist small pixel-wise discrepancies due to individual photon efficiencies and photon-to-voltage conversions. These differences can be subsumed to one process and can be thought of as a pixel-wise gain factor.

Dark Current Non-Uniformity (DCNU) Since the exact temperature varies across the sensor and also due to other inaccuracies, the amount of dark current induced per pixel will vary, and, more importantly, it will do so in a somewhat deterministic fashion, i.e., there will be some pixels that will generally suffer more from dark current than others. E.g., it is a well-known effect that the regions of the camera where the sensor is connected to the rest of the camera electronics tends to be hotter than other areas of the image sensor. This is known as *sensor glow*. This kind of noise can be thought of as pixel-wise bias.

Fixed Pattern Noise (FPN) Additionally, the read-out electronics behave non-uniformly, i.e. the noise added through the read-out process will also differ from pixel to pixel. The pattern that is introduced by the camera electronics alone, even in absence of dark current is also called *bias noise*. Sometimes, the term readout noise is also used interchangeably.

Although all spatial noise sources bias the overall noise effect from pixel to pixel, only the systematic effect of the readout noise will hereafter be referred to as bias noise. Also note that, although these terms are frequently used in the denoising community, they are often used slightly different from author to author. In the following, they will always be referred to in the terms in the above stated meaning.

Another major distinction between temporal and spatial noise is that, whereas temporal noise is an unavoidable limitation of the measurement process, the spatial noise is to some extent quantifiable and can therefore be reduced through careful calibration. In this sense, temporal noise sources are instances of the spatial pixel-wise deviations.

Additional to the above mentioned noise sources, there is also quantization noise, which occurs when an analog signal is converted into a digital signal of finite accuracy. However, in many cases this can be overcome by using enough bits for the input signal and using a loss-less image format. In practice, the number of bits needed for accurate digitalization can in principle be determined by the full-well capacity of the pixel, that is, the value of electrons that can be stored during exposure, and the conversion factor. Therefore, the accuracy of the image sensor could be improved upon, if necessary, and the other noise sources dominate the degradation significantly.

Good overviews of the different noise sources give, e.g., [25, 32, 33, 34, 35].

3.3 Camera Calibration

As with any physical metering instrument, good camera calibration is necessary to achieve best results. Many consumer cameras, specially modern DSLR (digital single-lens reflex) cameras, already apply built-in image denoising algorithms, if the user does not specifically save the data in raw, i.e., unprocessed, data format. On the other hand, many standard protocols and best practices have been developed over the years, especially within the astronomical community [25, 36], where camera noise from early CCD cameras has been most critical to imaging success.

In the following, the most important techniques and the respective terminology will be introduced and discussed where it is relevant to the rest of the thesis.

3.3.1 Bias Frames

It is insightful to consider the necessary conditions under which the different noise sources occur. PRNU can only occur when there is a light source to measure. Similarly, dark current will only have a significant impact during long enough exposure times such that the present thermal energy can free some electrons.

But even in complete absence of a light source and with the shortest possible exposure time, there will be inherent bias noise from the readout system. However, samples from this noise sources can readily be captured by taking images with a closed shutter and shortest possible exposure time. These images are called *bias frames*.

The pixel-wise mean or median of a set of bias frames is a good estimator for the readout noise in a digital camera and should always be subtracted from a recorded image, even if it was captured under good lighting conditions and with a short exposure time.

3.3.2 Dark Frames

Similar to bias frames, a *dark frame* is a sample of the dark current distribution in a digital camera. Since this is both temperature and time dependent, it is important to record dark frames under the same conditions as the actual image that is to be recorded, which is also called *light frame* or *science frame*. Additionally, it is to be expected that the dark current will change over the course of different imaging sessions, e.g., because of changing outside thermal conditions. Therefore, dark frames have to be recorded on a regular basis to have a good estimator of momentary sensor conditions.

Dark frames, like bias frames, are taken with closed shutter. Therefore, these kind of calibration images are sometimes referred to as *lens cap pictures*.

3.3.3 Flat-Fields

Finally, *flat-fields* are images of evenly distributed light sources. Flat-fields can therefore be used to measure relative change in measured pixel intensities for supposedly the same level of irradiance. Pixel-wise dividing a record image by the mean or median of a set of flat-fields will, in consequence, remove not only photo-response non-uniformities, but also other sources of non-regularities, such as the $\cos^4$-law, vignetting, or dust particles in the optics system. Conversely, since the latter will change with each application, it is necessary to capture flat-fields regularly and at least once for each configuration.

Although it is fairly straightforward to record flat-fields to measure instrument non-regularities, it is much harder to only capture the effects that is introduced by the imaging sensor itself without the surrounding optics. For this reason, this thesis focuses on the noise introduced by the readout and dark current noise.

3.3.4 Processing Protocols and Discussion

For practical purposes, the most important noise reduction mechanisms are *dark frame subtraction* and *flat-field division* [36], which is also sometimes called *flat-fielding*, applied in this order.

To generate a good flat-field, both samples of flat-fields, and, depending on the exact capturing procedure, bias or dark frames are necessary. The so-called master flat-field is usually the median of all recorded flat fields, each of which in turn get denoised with bias or dark frames.

In practice, there are essentially two different techniques for dark frame subtraction:

1. One can record a dark frame right before or after the capture of a light frame and use this as the best estimate for the current distribution of generated dark current.
2. One can record a set of dark frames for each configuration at the beginning or end of an recording session and use the mean dark frame as best estimate for the dark current in all the light images that were recorded during the operation.

It is easy to see that the second option is the maximum likelihood estimator for the denoised image, given that the conditions do not change over time [37]. Of course, also minor variations and combinations of this method are imaginable and more sophisticated algorithms have also been suggested [38, 39] which can also be thought of as belonging to the second category.

Both categories have advantages and disadvantages, which are in contrast to each other. Whereas the first category of methods will be more likely to capture the current condition of the image sensor, it is also less informed, since it only takes into account one measurement of the camera noise. If camera conditions are relatively stable of a longer period of time, the second category of camera calibration methods is to be preferred, since it will not only be more accurate due to more samples, but it will also take significantly less time, especially in astronomical settings, where exposure times can easily be 5-10 minutes and imaging conditions might change relatively fast due to weather conditions.

In Section 6.2, a detailed analysis based on recorded dark frames is presented and discussed.

4 Statistics of Natural and Astronomical Images

4.1 Motivation

As has been argued in Chapter 2, good assumptions and prior knowledge is key to successful image denoising. As long as the model is general enough to describe the real situation, it is beneficial to force the model towards the right direction, thus increasing the chance of finding the best possible solution and maybe even reduce the time needed to do so.

Reexamining the question stated in Section 1.2, the users of image denoising methods are in particular those photographers who have to deal with noise the most, i.e., photographers of content with a low signal-to-noise ratio. In usual lighting conditions, getting a good signal is not a problem. In fact, sometimes photographers are forced to artificially reduce the amount of light with a neutral density filter, if they want to use both long exposure and large aperture for artistic reasons.

On the other hand, images taken under poor lighting conditions usually suffer from a lot of noise. Although this is true for natural images taken at night or even at twilight, this thesis focuses specifically on astronomical images.

In astronomical imaging, exposure times are very long. Even with very good optics, exposure times are rarely below one minute, while typical exposures are 5-10 minutes. This will not only result in bias noise, but will also add a significant amount of dark current noise. To make matters worse, it is often the case that consumer cameras do not have a cooling system for the image sensor. Thus, the amount of dark current will even increase with ongoing exposure.

Although noise is an important problem in astro-photography, it also remains unclear whether traditional denoising methods work well on astronomical images. Most denoising algorithms focus on *natural image* and therefore incorporate statistics of natural images into the algorithms. In this context, the term natural images is not very well defined in literature. It refers to images from natural environments and day-to-day scenes. Basically anything that a human being could look upon in daily life or anything that could be captured with a camera.

However, there is an obvious problem with this definition: the problem of scale. Depending on the type of lens — normal, wide-angle or long-focus — the field of view of the image will be different from the angle-of-view of a human observer. For instance, objects will appear magnified in a long-focus lens and it is obvious that things look very different when looked at through a magnifying glass. The important question in this context is *whether the image statistics are scale-invariant.*

In traditional natural image denoising, however, this is not a real issue, since the data which generated the prior knowledge will most likely also have come from varying lenses and thus the expected statistics will match the calculated statistics. But looking through a telescope into space, this situation changes drastically. Although it is a naturally occurring scene, it is not something that humans could observe naturally. It is *non-natural image denoising.* Therefore, it has to be investigated whether the statistics of natural and astronomical images are different and if so, how they are different.

4.2 Statistical Comparison of Natural and Astronomical Images

It is intuitive that the statistics of astronomical images are very different from natural images. Whereas natural images cover a broad range of irradiance levels, most astronomical images tend to have either very low irradiance levels — the black sky or faint stars — or very high irradiance levels of bright stars. Secondly, the distributions of these values are also very likely to be highly different. In natural images, the distribution of pixel values is mostly uniform, whereas astronomical images are mainly black, i.e., zero and only rarely have a positive value. This is, the distribution of pixel values is *sparse.*

This can be seen by calculating the average L^p-norm for image patches and small values of p, where the L^p-norm is defined as:

$$L^p(x) = \left(\sum_{i=1}^{K} |x_i|^p \right)^{\frac{1}{p}} \qquad p \in (0, \infty) \tag{4.1}$$

$$L^0(x) = \sum_{i=1}^{K} \mathbf{1}_{\{x>0\}}(x_i) \tag{4.2}$$

where $\mathbf{1}_M$ denotes the indicator function of set M. It can be seen that the contribution of small pixel values to the average norm increases with smaller value of p, with $p = 0$ corresponding to the case, where the pixels with non-zero values are counted. Figure 4.1 shows the average norms of 10000 image patches extracted from different sets of benchmark images. A patch size of 21×21 was used, values for $p \in \{0, 1, 2\}$ have been calculated. Natural images were taken from the standard denoising benchmark set. SDSS and Capella images are described in more detail in Sections 4.3.1 and 4.3.2, respectively. From both data sets, a small set of images used for evaluation were set aside. The figure clearly shows that astronomical images are sparser than natural images.

Similarly, one can compare histograms of discrete image gradients. Figure 4.2 shows the log histogram for all three benchmark data sets. As expected, gradients in astronomical images are more concentrated around 0, corresponding to the large smooth areas. Secondly, the rest of the gradients is more evenly distributed on other values, since gradients of higher magnitude appear at edges of stars, which in turn can appear in any brightness. This can be seen by the heavier tails in the astronomical data.

Another method of comparing image patch statistics is to compare power spectrum signatures [40]. The results of this analysis can be seen in Figure 4.3. Two characteristics can be found:

1. Natural image patches are anisotropic (left panel), while astronomical images are isotropic (middle and right panel). This indicates that the distribution of image patches is more constrained in astronomical images, which will lead to simpler models.

2. The spectral signatures of astronomical image patches are broader, which further indicates sharp edges with smooth areas.

Finally, an important tool in image statistics is *Independent Component Analysis (ICA)*. In this setup, the signals in image patches are assumed to be the sum of independent signal sources. It has been found that describing images in this way leads to image patch representations that are likely also used by the visual cortex of humans [41]. Here, the Topographic ICA of [42] has been used. In this variant of ICA, residual dependency of independent components are modeled to define a neighborhood structure on the source signals, leading to visualizations which are easier to grasp.

Figure 4.4a depicts the independent components of 10000 natural image patches. Gabor-like filters can be found which have been observed both in image statistics analysis as well as unsupervised training of neural networks for image tasks. However, there are almost no Gabor-like filters in the independent components as can be seen in Figure 4.4b, one more indicator that the statistics of natural and astronomical images are fundamentally different.

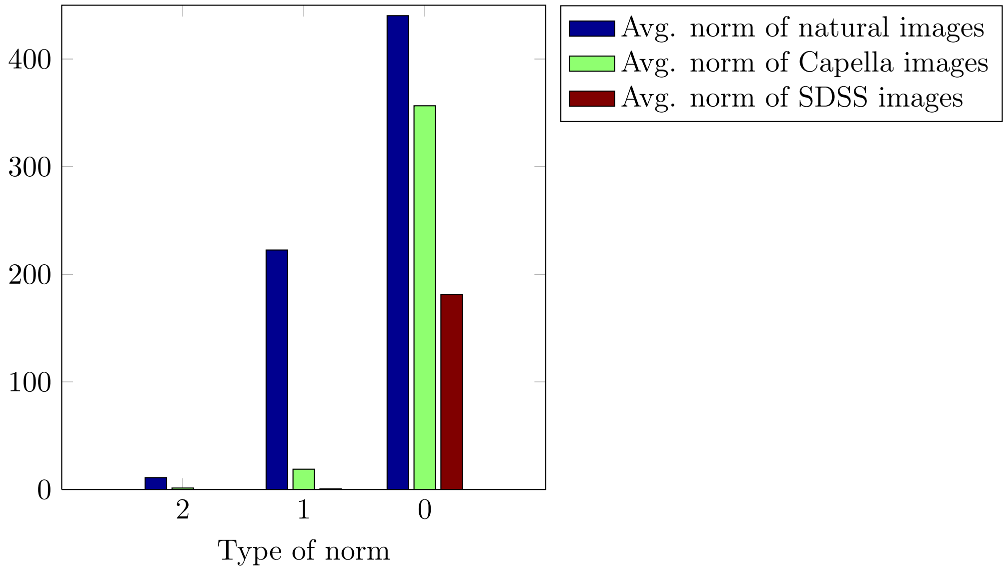


Figure 4.1: Average L^p-norm of image patches, extracted from different types of images. Values have been calculated for $p \in \{0, 1, 2\}$, where the L^0-norm is the counting measure. From each set of test images 10000 image patches of size 21×21 were extracted. See text for more details.

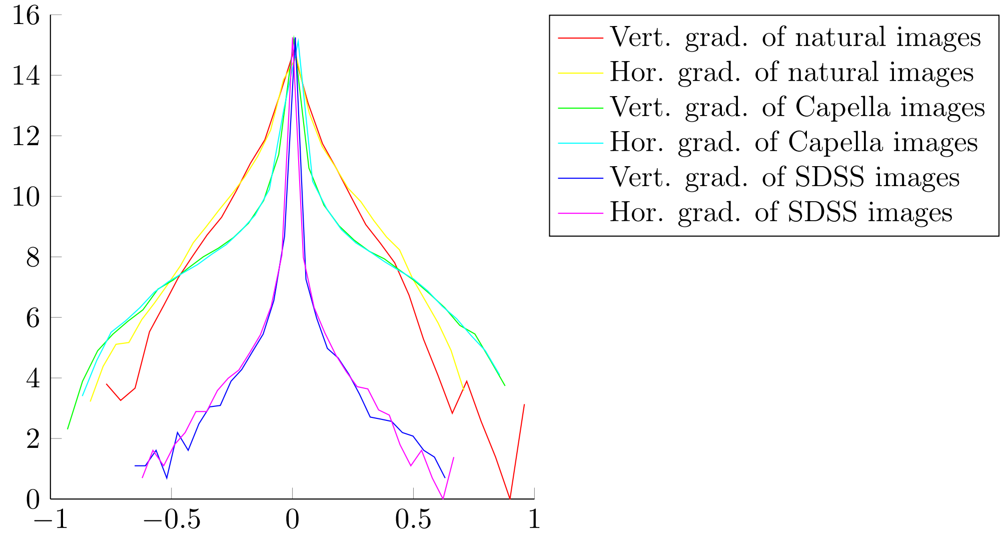


Figure 4.2: log-histogram of image patch gradients, extracted from different types of images. For each set of images, both horizontal and vertical gradients have been calculated on 10000 image patches of size 21×21. See text for more details.

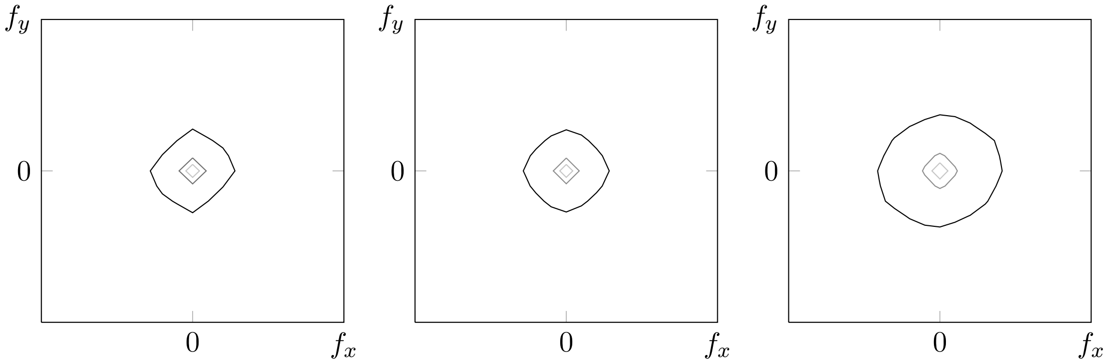


Figure 4.3: Power spectra of image patches, extracted from different types of images. Contour plots show 60%, 80% and 90% of the energy signatures (in log-scale). From left to right showing the signatures of: natural image patches, patches from Capella benchmark images, patches from SDSS benchmark images. For each spectra 10000 image patches of size 21 × 21 were extracted. See text for more details. See [40] for comparison.

4.3 Acquisition of Training Data Sets

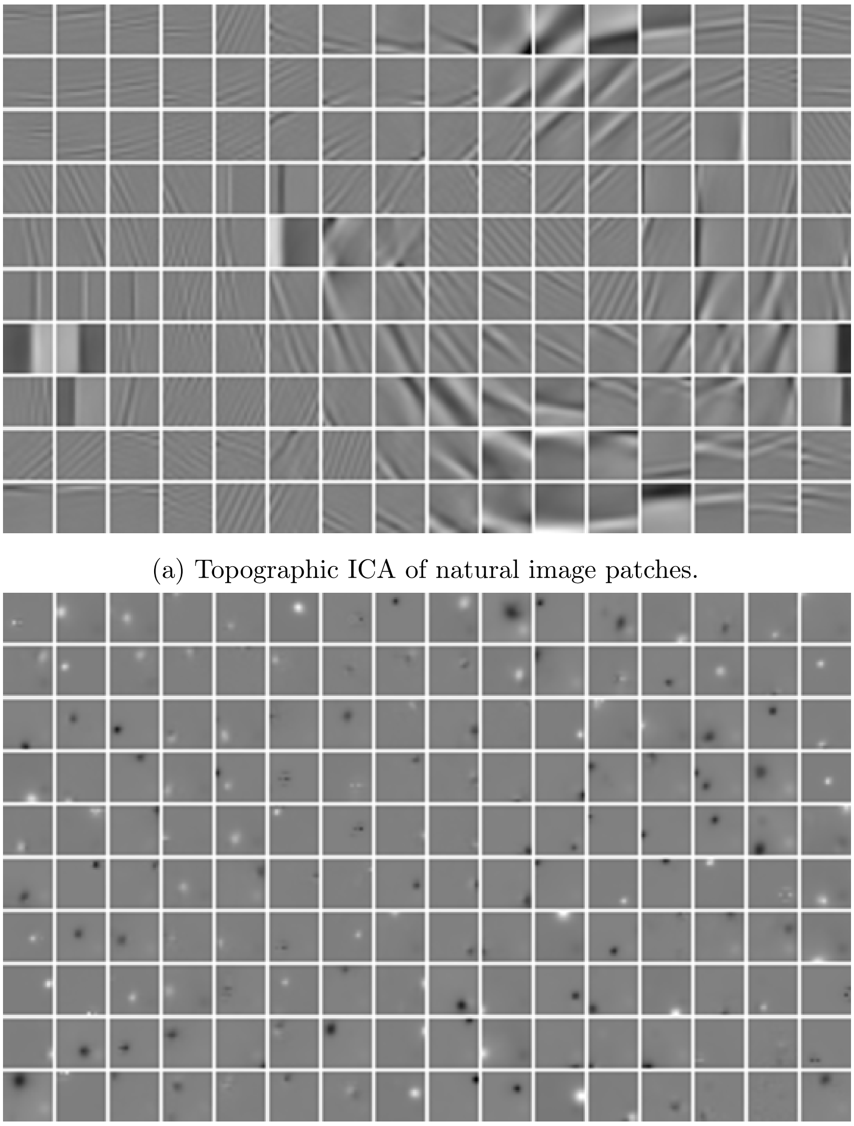
As has been argued in [9] and shown in [21], huge databases of natural images are needed to train machine learning algorithms for image denoising. This is not a problem for natural images, since training data is available in abundance on the Internet, both scientifically collected [43, 44] or simply on image sharing websites such as Flickr.

The situation changes if astronomical images are considered. This is for two reasons:

1. There are a lot less astronomical images available than natural images. Additionally, images with astronomical content come in many different shapes. For instance, NASA's *Astronomy Picture of the Day* may contain stars, galaxies, nebulaes as well as close-ups of planets or even spectrography images.

2. Astronomical images require much more post-processing after the capturing, thus making it difficult to build an automatic pipeline to ensure reasonable input to the learning algorithm.

Although the first issue could potentially be solved with an automated image classifier, the second issue is more pressing and more complicated. While natural images are rarely manually post-processed or only to a small extent, astronomical images are almost always post-processed by photographers. While denoising and flat-fielding are almost always done, common additional steps include stacking, gamma correction or other non-linear scaling and colorization for monochromatic



(a) Topographic ICA of natural image patches.

(b) Topographic ICA of astronomical image patches. Patches were taken from the SDSS benchmark data set.

Figure 4.4: Topographic ICA [42] for natural and astronomical image patches. 10000 patches of size 21 × 21 were extracted. Not shown are the independent components of the Capella data set, which are visibly very similar to the independent components of the SDSS image patches. See text for more details.

Fits images.

However, denoising is the first step in this pipeline. Therefore, to accurately represent the task, it is important to work with images that closely resemble the camera input/output rather than the „nice looking“ picture that will be eventually produced.

On the other hand, it is important to acquire mostly noise-free images for training, because otherwise the machine learning algorithm will produce noisy outputs after all. Therefore, some degree of post-processing might actually be desired to train a good denoising machine.

The following describes different image data sets that have been obtained and are used in the experiments.

4.3.1 Sloan Digital Sky Survey Data Set

The *Sloan Digital Sky Survey (SDSS)* [45, 46] is a large-scale scientific imaging project which covers a big percentage of the sky. It has been acquiring images and spectroscopies for several years, but is also an ongoing project. It uses a dedicated telescope at the Apache Point Observatory in New Mexico, USA.

Imaging is done with 30 CCD chips with five different filters applied in sequence on six parallel arranged in columns [47]. Images are recorded in drift scan leading to an effective exposure time of 54 seconds, while taking 71.7 from the first row of one chip to the first row of the next. The CCDs are highly cooled and very carefully manufactured, making the background flux of the sky the biggest noise source in the images.

The homepage of the SDSS provides an interface to download all captured images. Available are the FITS files for all filter bands as well as automatically generated preview-JPEGs, which incorporate the g, r and i band of the recordings. To ensure relevant content on the images, all available objects of the New General Catalogue have been downloaded, as well as a random selection of fields. In the experiments, the data from the FITS images of the g, r and i bands were used.

There are both, potential benefits and short-comings of the SDSS data. An obvious advantage is the availability of much little-processed data, which, at the same time, has been carefully recorded by professional astronomers. Therefore, the raw data is likely to be a very close representation of ground truth camera recordings. A downside is that there is only one capturing setup, i.e., one fixed aperture and pixel resolution, which might limit the generalization of the trained model. Secondly, since these images are recorded fully automatic, some images suffer defects, which would require manual selection at a scale that is not applicable. Figure 4.5 shows a clearly corrupted

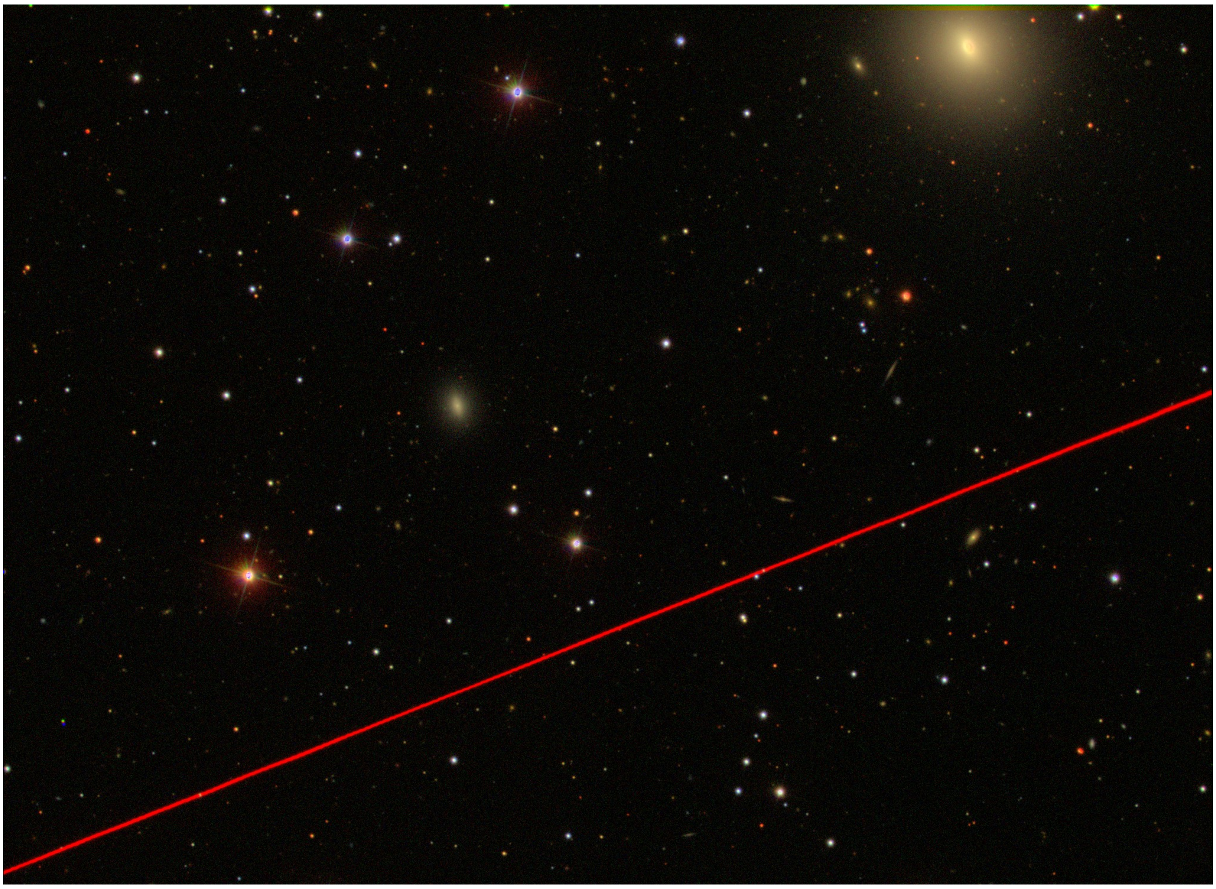


Figure 4.5: An example of an image recorded from the Sloan Digital Sky Survey. Artifact is clearly visible.

image. Here, the red straight line is likely caused by an airplane crossing the field of view, but other defects are also observed.

4.3.2 Capella Observatory Data Set

Additional to the SDSS data set, images from the Capella Observatory website [48] are used. The Capella Observatory is a privately run observatory on Crete, Greece. Although many images are recorded using the main instrumentation of the observatory, the website also features some images taken by the authors on different occasions with other equipment, making the data set more versatile than the SDSS data set.

However, the available data is completely post-processed, i.e., all filtering steps have already been applied. Although this makes for visually pleasing images, the post-processing cannot be easily undone automatically, which might have an effect on the trained denoising model. Furthermore, the available data is in a lossy image format suitable for the Internet, which might further affect denoising performance. Secondly, there is only a small amount of images available as compared to other image data sets.

5 Methods

5.1 Camera Noise Model

In its general problem formulation, image denoising can be viewed both as a signal processing problem, leading to more engineered algorithms, or as a problem of statistics, leading to learning-based methods. In this thesis, the goal is to apply image denoising to a subset of the general setup where denoising is both particularly important and much more specific. As has been argued in Chapters 3 and 4, the statistical properties of the general case do not transfer to this problem. Therefore, methods which are capable to adjust to the restricted probability distributions are expected to outperform methods which cannot take this additional source of information into account.

However, in order to extract the necessary information, one needs access to appropriate training data. In the case of real cameras and astronomical images, this is a major problem for several reasons:

- It is difficult to capture the exact same scene both with a noisy and a noise-free, but otherwise identical, setup.
- Averaging is not a good option, since the imaging conditions for astro-photographs are also subject to other image degradations, such as poor seeing or tracking errors, which should consequently be treated independently and not influence the denoising procedure.
- It is not feasible to acquire a big enough training data set for each camera individually with astronomical images, since recording under good conditions is too time consuming for practical purposes.

Therefore, other ways of generating a training database are required.

In accordance with existing literature on camera noise removal [32, 33, 34] as well as the AWGN model, an additive noise model will be assumed in this thesis. Furthermore, this thesis focuses on dark current noise and read-out noise, since those are the most urgent noise sources for astronomical images. Consequently, a noisy observation y_i of a clean image x_i at pixel i is given by:

$$\begin{aligned} y_i &= x_i + df_i && (5.1) \\ &= x_i + dc_i + b_i && (5.2) \end{aligned}$$

where df_i in (5.1) is the dark frame noise at pixel i, which in turn is the sum of dc_i and b_i, i.e., the dark current noise and bias noise at pixel i, respectively.

Although the exact distributions of dc_i and b_i are unknown, samples from the distribution of df_i can be drawn by taking dark frame images, as has been mentioned in Section 3.3.2, which corresponds to

$$\begin{aligned} y_i &= x_i + df_i && (5.3) \\ &= 0 + df_i && (5.4) \\ &= df_i && (5.5) \\ &= dc_i + b_i && (5.6) \end{aligned}$$

In the case of the shortest possible exposure time, it can further be assumed that $dc_i = 0$. Recall from Section 3.3.1 that such images are called bias frames.

From a recorded set of bias frames and dark frames, it is possible to calculate the average amount of noise on an actual image by taking the sample mean. That is, for each image and each pixel, the noise can always be described as deviation of the mean noise added:

$$\begin{aligned} df_i &= \overline{df_i} + df_i^{res} && (5.7) \\ dc_i &= \overline{dc_i} + dc_i^{res} && (5.8) \\ b_i &= \overline{b_i} + b_i^{res} && (5.9) \end{aligned}$$

where the bar $\overline{n_i}$ over a variable n denotes the empirical sample mean and n_i^{res} will be called the *residual error* (of noise type n) and is in fact a random variable of its own right.

After subtraction of the average dark frame $\overline{df_i} = \overline{dc_i} + \overline{b_i}$, the noisy observation is given by

$$y_i = x_i + dc_i^{res} + b_i^{res} \qquad (5.10)$$

where dc_i^{res} and b_i^{res} are random variables with zero mean.

Given the fact that the sum of two independent random variables X and Y each drawn from not necessarily equal Gaussian distributions $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ and $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$ is itself again

Gaussian and distributed according to

$$X + Y \sim \mathcal{N}(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2) \tag{5.11}$$

means that $n_i = dc_i^{res} + b_i^{res}$ has a distribution of $\mathcal{N}(0, \sigma^2)$ for all $i \in \{1, \ldots, N\}$, if all dc_i^{res} and b_i^{res} are distributed according to a Gaussian distribution and for all i, it further holds

$$\sigma^2 = \sigma^2_{dc_i^{res}} + \sigma^2_{b_i^{res}} \tag{5.12}$$

Therefore, Equation (5.10) both shows the relationship to the AWGN model, as well as the potential shortcomings of it. Because, in order for Equation (5.10) to be in form of the general noise model (2.2), Equation (5.12) must hold.

Of course, many of these assumptions are highly unlikely for actual cameras. E.g., it is plausible that a pixel with a high variance in its bias noise will also have a high variance in its dark current. Therefore, the sum of these noise sources will probably *not* be independent. Furthermore, it is unlikely that (5.12) will hold for all pixels.

And finally, it is unlikely that all of the involved distributions are Gaussian to begin with. For instance, this is obviously not true for *„hot pixels“* or *„dead pixels“*. The former correspond to a very high sensitivity to dark current and these pixels record the maximum signal $y_i = I_{\max}$ for each captured image. Dead pixels, on the other hand, do not record any signal at all and have fixed output of $y_i = 0$. Both phenomenon are well known artifacts among photographers and many software tools add automated correction for these issues, if a list of stuck pixels is known [49].

Another source of noise that must be considered, is the effect of *pixel saturation*. Saturation occurs when the electron well of a pixel is filled but still more electrons are freed (either by arriving photons or more dark current) and cannot be collected. In this case, the measured signal at location i will be the maximum possible value $I_{\max}$ of digital units for a given camera. Although this is not a problem for the signal input — if saturation occurs for a desired object, the exposure time can be reduced until saturation is no longer reached —, it is a problem, if a pixel is very noisy, i.e., $\overline{df_i} > I_{\max}$, in which case the original signal x_i at pixel i must completely be restored from its surrounding pixels.

Since subtracting the dark frame is both the maximum likelihood estimator for an individual pixel [37], as well as an important calibration technique in signal processing generally, the reported signal value will be systematically underestimated.

5.1.1 Generating Training Data for the Camera Noise Model

In case of additive white Gaussian noise, a lot of training data is readily available by generating random samples from the desired noise distribution and taking large image data sets which are available online, either collected specifically for scientific purposes [43, 44] or simply by large image sharing websites.

Similarly, samples from realistic camera noise can be generated, as has been explained in Section 5.1, and images can be downloaded from various sources, which has been detailed in Section 4.3.

A naive way of generating a noisy image would be to simply add a dark frame to a clean image:

$$y_i = x_i + df_i \tag{5.13}$$

where df_i comes from a set of generated dark frames. However, this assumes the wrong dynamical range of the camera.

First of all, it is necessary to scale input images to the same range $[0, I_{\max}]$. The cameras that will be used in the experiments have analog-to-digital converters with 14 and 16 bits respectively. Common image formats for sharing on the Internet have only 8 bits. Scientific images as produced by the Sloan survey have 10 bits.

Secondly, while digital units as reported by cameras are linear transformations of the measured physical signals, many processed digital image formats save data by applying *gamma correction*, which non-linearly transforms the data to capture more dynamic range in regions more sensitive to the human observer.

But most importantly, since the original noise in the camera has not a zero mean, but a positive offset, while the signal is also strictly positive, not the full dynamic range of the analog-to-digital converter is available to the signal. Additionally, camera manufacturers sometimes add a positive offset b to each pixel to allow for negative read-out noise instances. Assuming a dynamic range of $[0, I_{\max}]$ will therefore lead to clipping of the signal whenever $y_i > I_{\max} - b$.

Therefore, the camera noise will be generated as the following:

From a set of bias frames $\{b_i^t \,|\, i = 1, \ldots, N,\ t = 1, \ldots, T\}$, the minimum measured value

$$\hat{b} = \min_{\substack{i=1,\ldots,N, \\ t=1,\ldots,T}} b_i^t \tag{5.14}$$

is taken to be an estimator for the true offset b. Clean images are scaled to be in $[0, I_{\max} - \hat{b}]$ such that if no noise is present and the unmodified clean image is presented, they exactly fit into the dynamic range of the camera. Then, a clean image and a dark frame are added, but clipped to the maximal signal, as in

$$y_i = \min\left(x_i + df_i, I_{\max}\right) \tag{5.15}$$

To give a fair comparison with other denoising algorithms, the mean dark frame is removed, in order to get as close to the general problem formulation as possible:

$$\widehat{y_i} = \min\left(x_i + df_i, I_{\max}\right) - \widehat{df_i} \tag{5.16}$$

This processed image $\hat{I} = \{\widehat{y_i} \mid i = 1, \dots, N\}$ is given as input to the denoising algorithm.

5.2 Using Patch-based Denoising Algorithms

Speaking from a machine learning perspective, the task of denoising an image can be viewed as a general *regression task*, that is, of finding a mapping $f : \widehat{y_i} \to \widehat{x_i}$, where the output $\widehat{x_i}$ has continuous support. A wide variety of machine learning algorithms have been proposed to solve problems of this kind. Recently, *Multi-Layer Perceptrons (MLP)* have been applied successfully to general image denoising [7, 21] and are therefore a straightforward choice to also apply to camera-specific image denoising.

Although there is no formal restriction to the input and output dimensions of a MLP such that a one-shot prediction of the whole image

$$f : [0, I_{\max}]^N \to [0, I_{\max}]^N \tag{5.17}$$

$$f(\{\widehat{y_i}\}_{i=1}^N) = \{\widehat{x_i}\}_{i=1}^N \tag{5.18}$$

is imaginable, it is infeasible in practice, both due to computational complexity as well as the curse of dimensionality: in order to generalize, a MLP must learn the full joint distribution of pixel values whose size grows exponentially with the number of dimensions. Therefore, an exponential number of training images would be needed to create a potentially good generalization.

A solution that has been found to work well in practice is to find estimators for small image patches [50, 1, 21], whereby an image is divided into small patches of equal size such that each pixel belongs to at least one patch. Patches are usually quadratic and may overlap, in which case there are multiple predictions for some pixels, allowing for some method of producing a final estimate from these predictions. In most cases, individual estimates will be averaged.

The size K of the input patch size needs not be equal to the size k of the output patch. In fact, choosing k small will greatly reduce the complexity of the learning problem, although it might be helpful to maintain a slightly larger output size to leverage some gain from averaging estimates. And secondly, there is also an important speed-up to be gained by choosing a larger k, since patch offsets greater than 1 reduce the number of patch predictions quadratically.

On the other hand, there is a trade-off in the parameter of the input patch size. Choosing a larger patch size K will make the learning problem more difficult, while at the same time constraining the space of possible explanations for a given input patch by adding context. It has been shown both theoretical [51, 52] as well as empirical [53], that in the case for AWG noise larger input patch sizes are required with increasing σ.

While this certainly holds for natural image patches, it will be a question of this thesis whether this also holds for astronomical image patches, since stars are spatially very limited features on these images and nebulae are just principally diffuse objects.

5.3 Multi-Layer Perceptrons as Image Patch Denoising Algorithms

5.3.1 Perceptrons and Multi-Layer Perceptrons as General Regression Models

A *perceptron* is a linear function [1]

$$f : \mathbb{R}^K \to \mathbb{R}^k \tag{5.19}$$

$$f(x) = Wx + b \tag{5.20}$$

where $W \in \mathbb{R}^{k \times K}, b \in \mathbb{R}^k$, which can either be used as classifier or regressor. Finding the *optimal* parameters for the problem setting at hand depends on the desired output, i.e., on the evaluation metric and its associated *loss function*, and on the available and expected input data. The word „optimal“ has to be used with caution in this context, because many parameter pairs could potentially lead to equally good descriptions of the data.

It is common in the image denoising community to evaluate image enhancement algorithms based on the Peak Signal-to-Noise Ratio PSNR between the output of the algorithm and the known ground truth clean image. Since this function depends monotonically on the mean squared

[1] In contrast to previous sections, the input of the functions will be denoted by x and outputs will be denoted by $\hat{y}$, which partially reverses the conventions of the previous chapters, but is more traditional for the analysis of multi-layer perceptrons.

error, optimizing the perceptron with a quadratic loss function

$$\mathcal{L}(\hat{y}, y) = \parallel \hat{y} - y \parallel_2^2 \tag{5.21}$$

$$\mathcal{L} = \frac{1}{S} \sum_{s=1}^{S} \mathcal{L}(\hat{y_s}, y_s) \tag{5.22}$$

where $\hat{y_s} = f(x_s)$, y_s is the true value and $\mathcal{D} = \{(x_s, y_s) \mid s = 1, \ldots, S\}$ is the training data set, will also optimize the PSNR results.

However, denoising image patches is highly unlikely to be a linear problem and in this case a simple perceptron is not expected to perform well. Therefore, a non-linear generalization of the perceptron is required. This has classically been the *multi-layer perceptron* which is simply a combination of linear and non-linear functions applied in sequence, e.g.,

$$f(x) = b_3 + W_3 \tanh(b_2 + W_2 \tanh(b_1 + W_1 x)) \tag{5.23}$$

where the tanh-function is to be applied element-wise.

The further analysis will be simplified with some additional notation. A MLP $f : \mathbb{R}^K \to \mathbb{R}^k$ is given by a sequence of functions $f_i : \mathbb{R}^{k_{i-1}} \to \mathbb{R}^{k_i}$, $i \in \{1, \ldots, l\}$, $f(x) = (f_l \circ \cdots \circ f_1)(x)$ each of which has a set of parameters $\Theta_i = \{\theta_{i,j} \mid j = 1, \ldots, n_j\}$. Θ_i may also be empty if $n_i = 0$. $\Theta = \bigcup_{i=1}^{l} \Theta_i$ denotes the set of all parameters and $\theta = (\theta_{1,1}, \ldots, \theta_{1,n_1}, \ldots, \theta_{l,1}, \ldots, \theta_{l,n_l})$ denotes a vector consisting of all parameters in all functions. The result of the application up to the first i functions will be denoted as $x_i = (f_i \circ \cdots \circ f_1)(x)$ which includes $x_0 = x$, $x_l = \hat{y}$ and $x_i = f_i(x_{i-1})$ as special cases.

To illustrate this notation, the MLP (5.23) can be rewritten as

$$f(x) = (f_5 \circ f_4 \circ f_3 \circ f_2 \circ f_1)(x) \tag{5.24}$$

$$f_5(x_4) = b_5 + W_5 x_4 \tag{5.25}$$

$$f_4(x_3) = \tanh(x_3) \tag{5.26}$$

$$f_3(x_2) = b_3 + W_3 x_2 \tag{5.27}$$

$$f_2(x_1) = \tanh(x_1) \tag{5.28}$$

$$f_1(x_0) = b_1 + W_1 x_0 \tag{5.29}$$

where $\Theta_i = \{(W_i, b_i) \mid i \in \{1, 3, 5\}\}$ and $\Theta_i = \emptyset$ for $i = 2, 4$.

It has been shown [54, 55, 56] that multi-layer perceptrons are universal function approximators, i.e., given any function $g : \mathbb{R}^K \to \mathbb{R}^k$, there exists a multi-layer perceptron $\hat{f}$ such that the error between $\hat{f}$ and g is arbitrarily small. However, there is no tractable way to determine the form of $\hat{f}$ or its parameters. Therefore, heuristics are needed to find a combination of parameters which work well in practice.

5.3.2 Training of Multi-Layer Perceptrons: Back Propagation

Training multi-layer perceptrons is not a convex optimization problem, and therefore no tractable methods exist for finding a global optimum so far. A heuristic, that has to be proven to work very well in practice, is to use a sort of *gradient descent*.

Gradient descent starts with an initial guess for all parameters θ^0 and updates the parameters according to

$$\theta^{t+1} \leftarrow \theta^t - \lambda \frac{\partial \mathcal{L}}{\partial \theta} \tag{5.30}$$

with *learning rate* λ, typically taking values $\lambda \in (0, 1)$.

In the case of large training data sets, it is not feasible to compute the exact gradient of the mean squared error (5.22), since this would require to compute the quadratic error for all possible input training patches. Instead, the full gradient is computed approximately by taking the derivative only with respect to one training patch or a small selection of training patches. Both cases are instances of the so-called *stochastic gradient descent*, where the latter is referred to as *mini-batch stochastic gradient descent* and the former is called *single (instance) stochastic gradient descent* or simply stochastic gradient descent.

A difficulty remains in computing the gradient $\frac{\partial \mathcal{L}}{\partial \theta}$, even for the case of a single training instance. However, it turns out that this can be computed efficiently using simple rules of calculus, namely the chain rule for derivatives, and dynamic programming, where the application of both combined is called *back propagation*.

To see how it works, one can first consider the case where only the parameters of the last layer should be updated. For a single training instance x and its current prediction $\hat{y}$, the gradient for each parameter in the last layer can be expressed as

$$\frac{\partial \mathcal{L}}{\partial \theta_{l,j}} = \frac{\partial \mathcal{L}}{\partial f_l} \frac{\partial f_l}{\partial \theta_{l,j}} \qquad \forall j = 1, \dots, n_l \tag{5.31}$$

$$= \frac{\partial \mathcal{L}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial \theta_{l,j}} \qquad \forall j = 1, \dots, n_l \tag{5.32}$$

The first factor of (5.32) is known since $\frac{\partial \mathcal{L}}{\partial \hat{y}} = 2(\hat{y} - y)$. The second factor depends solely on the last function that has been used in the sequence. Therefore, if we use functions with tractable derivatives, the calculation becomes easy. E.g., in the case of (5.24), one gets

$$\frac{\partial \hat{y}}{\partial W_5} = (\tanh(b_3 + W_3 \tanh(b_1 + W_1 x)))^T \tag{5.33}$$

$$= x_4^T \tag{5.34}$$

$$\frac{\partial \hat{y}}{\partial b_5} = 1 \tag{5.35}$$

From (5.34) it becomes clear where the dynamic programming needs to be applied, since the right hand side is basically a value that has already been computed along the way for the evaluation of $\hat{y}$. Therefore, if the intermediate value has been stored previously, evaluating the gradient in the last layer is a simple calculation.

With further applications of the chain rule, it also follows what the gradient of the second last layer should be:

$$\frac{\partial \mathcal{L}}{\partial \theta_{l-1,j}} = \frac{\partial \mathcal{L}}{\partial f_l} \frac{\partial f_l}{\partial f_{l-1}} \frac{\partial f_{l-1}}{\partial \theta_{l-1,j}} \qquad \forall j = 1, \ldots, n_{l-1} \tag{5.36}$$

$$= \frac{\partial \mathcal{L}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial f_{l-1}} \frac{\partial f_{l-1}}{\partial \theta_{l-1,j}} \qquad \forall j = 1, \ldots, n_{l-1} \tag{5.37}$$

As was the case for the last layer, it is required that derivatives of layer $l-1$ with respect to its parameters are tractable. Furthermore, if the computation of the derivatives of layer l with respect to its input from layer $l-1$ are also tractable, the whole gradient becomes tractable with respect to the parameters in layer $l-1$. For instance, in the case of (5.24), it can be seen that

$$\frac{\partial \mathcal{L}}{\partial W_3} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial f_4} \frac{\partial f_4}{\partial f_3} \frac{\partial f_3}{\partial W_3} \tag{5.38}$$

$$= \left[(1 - \tanh(x_3)^2) \odot W_5^T 2(\hat{y} - y)\right] x_2^T \tag{5.39}$$

$$\frac{\partial \mathcal{L}}{\partial b_3} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial f_4} \frac{\partial f_4}{\partial f_3} \frac{\partial f_3}{\partial b_3} \tag{5.40}$$

$$= \left[(1 - \tanh(x_3)^2) \odot W_5^T 2(\hat{y} - y)\right] 1 \tag{5.41}$$

where tanh is applied element-wise and $\odot$ donates the Hadamard product of element-wise multiplication. Note, that in both Equations (5.39) and (5.41) the first factor is the composition of its previous steps written „outside in“, but the individual steps can still be seen. The application of the tanh-function is considered to be its own layer, which is usually not parametrized, and therefore only passes a new gradient to its previous layer.

Although Equations (5.39) and (5.41) seem to be computationally expensive, they are in fact not more difficult to compute than the gradients in the last layer, if each layer computes its gradient with respect to the layer input and passes it to its predecessor. In the case of (5.39), the criterion layer will compute $2(\hat{y} - y)$ and pass it as $\frac{\partial \mathcal{L}}{\partial x_5}$ to the f_5 layer. The f_5 layer will compute $W_5^T \frac{\partial \mathcal{L}}{\partial x_5}$, and pass it as $\frac{\partial \mathcal{L}}{\partial x_4}$ to the f_4 layer and so forth. In each step, each individual layer relies on its input of this instance and the gradient from its successor of this instance. In turn, each layer provides its successor with its output and its predecessor with its gradient. Figure 5.1 illustrates the information flow from layer to layer in a MLP.

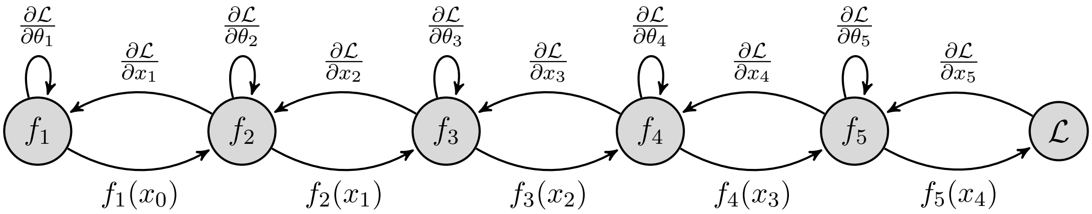

$\frac{\partial \mathcal{L}}{\partial \theta_1}$
$\frac{\partial \mathcal{L}}{\partial x_1}$
$\frac{\partial \mathcal{L}}{\partial \theta_2}$
$\frac{\partial \mathcal{L}}{\partial x_2}$
$\frac{\partial \mathcal{L}}{\partial \theta_3}$
$\frac{\partial \mathcal{L}}{\partial x_3}$
$\frac{\partial \mathcal{L}}{\partial \theta_4}$
$\frac{\partial \mathcal{L}}{\partial x_4}$
$\frac{\partial \mathcal{L}}{\partial \theta_5}$
$\frac{\partial \mathcal{L}}{\partial x_5}$
f_1
f_2
f_3
f_4
f_5
$\mathcal{L}$
$f_1(x_0)$
$f_2(x_1)$
$f_3(x_2)$
$f_4(x_3)$
$f_5(x_4)$

Figure 5.1: An illustration of the information flow in the Backpropagation Algorithm

In summary, in order to tractably and efficiently compute the gradient with respect to all parameters of all layers, it is sufficient to be able to compute the gradient of each intermediate function with respect to its parameters as well as with respect to its input. Furthermore, the gradient can be calculated layer-wise. However, there is a small overhead of storing the previous input to each layer, although his is not an issue in practice.

Furthermore, the general framework is not restricted to linear layers and tanh-layers. In fact, as long as the computation of the derivatives with respect to the layer parameters and input parameters is tractable, any kind of non-linear layer can be used. Table 5.1 gives an overview of the most common transfer functions.

Name	Transfer function	Derivative
Logistic	$\frac{1}{1+e^{-x}}$	$\frac{e^{-x}}{(1+e^{-x})^2}$
Tanh	$\tanh(x)$	$1 - \tanh(x)^2$
Rectified Linear	$\max(0, x)$	$\mathbf{1}_{\{0 \leq x\}}$
Ramp	$\max(-1, \min(x, 1))$	$\mathbf{1}_{\{-1 \leq x \leq 1\}}$

Table 5.1: A list of various non-linear functions used in multi-layer perceptrons with its transfer function and derivatives.

5.3.3 Practical Considerations for Training of Multi-Layer Perceptrons

Although the framework derived in the previous section is very general and applicable in many situations, a set of precautionary measures must be taken to increase the likelihood of convergence of the parameters to a good minimum. This holds even more so, since it is by no means guaranteed that the energy landscape of the loss function is convex or otherwise favorable for the task.

In fact, although MLPs have been around for a long time now, research interest has seen multiple highs and lows since the course of its discovery. This has several reasons. On the one hand, MLPs are computationally expensive. Although the complete basic ideas and algorithms can be written down in less than five pages, applying the methods to actual data is both memory and computationally expensive. This has certainly been a problem in the high days of MLPs in the late eighties and early nineties, but this is true even today. MLPs that are very successful at important practical problems like vision and speech processing are huge. A recent state-of-the-art algorithm on the established ImageNet data set used 650.000 neurons and 60 million parameters [57]. The input is 150,528-dimensional and it uses over eight layers. In order to have any chance of training an architecture of this size, very efficient implementations on modern GPU hardware is needed and training takes several days.

Secondly, and more importantly, there are a number of „tricks of the trade“ [58], meaning precautions and pre-processing considerations that must be taken into account while setting up the basic framework and problem specification. These preparations increase the chance of getting good convergence rates and generalization results, but cannot offer guarantees. Yet, not following these considerations, it is easy to see that problems during training are to be expected.

In order for gradient descent to find a good solution, it is key to provide conditions such that the gradients can explore a reasonable amount of the search space. This requires two courses of action:

1. Initial conditions need to be favorable for good exploration.

2. Gradients need to be both big enough to cover the space and small enough to converge to good local minimum.

Therefore, the first set of considerations deal with good initial conditions of the training. To this category belong:

Data normalization Both input and output variables are scaled linearly to be approximately Gaussian distributed, i.e., input and output variables are subtracted with their respective means and likewise divided by their standard deviations such that at least the first and second moment of the corresponding distributions match the Gaussian distribution.

Weight initialization Initial weights are not chosen completely random but are systematically drawn from a uniform distribution given by

$$\theta^0_{i,j} \sim \mathcal{U}\left[-\frac{\sqrt{6}}{\sqrt{k_{i-1}+k_i}}, \frac{\sqrt{6}}{\sqrt{k_{i-1}+k_i}}\right] \tag{5.42}$$

Choice of nonlinearity If a sigmoidal function is to be chosen, use the tanh-layer instead of the logistic layer, which is point symmetric and therefore better preserves mathematical properties between layers.

Combining this three methods ensures that both the linear and nonlinear part of the nonlinear layer is reached as well as having somewhat similar initial conditions for each layer [59].

Secondly, there are measures to adopt good search steps and directions.

One problem of stochastic gradient descent is that the computed directions can vary wildly, resulting in a very zig-zaggy convergence, which can be very slow. A possible solution to this dilemma is to keep information from previous gradient evaluations:

$$\Delta^{t+1}_{\theta_{i,j}} \leftarrow \nu\Delta^{t}_{\theta_{i,j}} - \frac{\partial\mathcal{L}}{\partial\theta^t_{i,j}} \tag{5.43}$$

$$\theta^{t+1}_{i,j} \leftarrow \theta^t_{i,j} + \lambda\Delta^{t+1}_{\theta_{i,j}} \tag{5.44}$$

In this formulation, each parameter gets an additional update variable called *momentum term*, which keeps track of previous updates in this dimension. An intuition of the search direction is the trajectory of a ball rolling down a hill, which also carries momentum — hence the name. In practice, this smooths the search trajectory for the parameter making convergence more stable.

Both, stochastic gradient descent as well as the variant with momentum, need to adjust the learning rate over time, to prevent from continuously skipping over a local minimum in its proximity. Common practice is to either continuously decrease the learning rate or initialize the search with a big learning rate and fine tune with a small learning rate. Furthermore, the learning rate is divided by the size of the input dimension k_{i-1} for each layer.

Recently, other methods of adaptive descent algorithms have been suggested which also adjust step size and directions over time. In this thesis, two different variants will be tested: *ADAGRAD* [60] and *ADADELTA* [61].

The key insight in both algorithms is to not only keep previous search directions, but also adjust the step size on previous gradient evaluations. In this thesis, a slightly modified version of ADAGRAD will be used:

$$g_{i,j}^{t} \leftarrow \frac{\partial \mathcal{L}}{\partial \theta_{i,j}^{t}} \tag{5.45}$$

$$\Delta_{\theta_{i,j}}^{t+1} \leftarrow -\frac{1}{\sqrt{\sum_{r=1}^{t}(g_{i,j}^{r})^2 + \epsilon}} g_{i,j}^{t} \tag{5.46}$$

$$\theta_{i,j}^{t+1} \leftarrow \theta_{i,j}^{t} + \lambda \Delta_{\theta_{i,j}}^{t+1} \tag{5.47}$$

where ϵ is a small constant to not divide by zero in the first step. Although it still uses a learning rate parameter λ, it is usually much more robust to the particular choice of λ.

However, due to its infinite history, the size of the updates will also decay fast, which will result in long running times until complete convergence. An improvement to this update rule is the ADADELTA algorithm, which utilizes a decay parameter δ to „forget“ earlier gradients:

$$g_{i,j}^{t} \leftarrow \frac{\partial \mathcal{L}}{\partial \theta_{i,j}^{t}} \tag{5.48}$$

$$\Delta_{\theta_{i,j}}^{t+1} \leftarrow -\frac{\sqrt{\sum_{r=1}^{t-1} \delta^{r} (\Delta_{\theta_{i,j}}^{r})^2 + \epsilon}}{\sqrt{\sum_{r=1}^{t} \delta^{r} (g_{i,j}^{r})^2 + \epsilon}} g_{i,j}^{t} \tag{5.49}$$

$$\theta_{i,j}^{t+1} \leftarrow \theta_{i,j}^{t} + \lambda \Delta_{\theta_{i,j}}^{t+1} \tag{5.50}$$

This update algorithm has yet more parameters, but has been shown to be both robust and very fast in practice.

Finally, there is one more trick that will be evaluated in this thesis: since the distribution of astronomical image patches is very sparse, the gradients are expected to be distributed very unevenly. Additionally, as will be seen in Section 6.2, the residual noise will be very small as compared to the signal, making the problem mostly linear with only a few exceptions. Both properties might be problematic for gradient descent.

However, we can inverse these properties by not focusing on the signal, but try to predict the noise instead, i.e., the estimator for the clean image patch is given by [2]

$$f(\hat{y}) = \widehat{df^{\,res}} \tag{5.51}$$

$$\hat{x} = \hat{y} - \widehat{df^{\,res}} \tag{5.52}$$

[2] Making the connection to the previous sections, the following formulas are presented in the regular denoising notation

Although it is very unusual to learn the defect instead of the desired goal, it might be a useful representation in this setting, because gradients will be stronger and more homogeneous during training, making it easier to find good local minima. No special property of image denoising is used in this trick, but rather the fact that the goal is to solve an under-constrained equation system of two variables with one equation. Therefore, this technique could potentially also generalize to other similar problem settings.

6 Experiments

6.1 Camera Equipment

6.1.1 Moravian G2-8300

The main camera used for experiments in this thesis is a Moravian G2-8300 with a Kodak KAF-8300 full-frame CCD chip. The camera also has a built in cooling system for the CCD chip to provide a controlled image capturing environment suitable for scientific and low-light applications [62]. Table 6.1 shows some of the technical properties of this camera.

Property	Value
Resolution	3358 × 2536 pixels
Pixel size	$5.4 \times 5.4\ \mu m$
Imaging area	$18.1 \times 13.7\ mm$
Full well capacity	Approx. $25\,000\, e^-$
Output node capacity	Approx. $55\,000\, e^-$
Dark current	$0.15\ e^-/s/$pixel at $0\,°C$
Dark signal doubling	$5.8\,°C$
ADC resolution	16 bits
Gain	$0.4\ e^-/$ADU
System read noise	$9\ e^-$
Maximal ΔT	$> 50\,°C$ below ambient
Regulation precision	$\pm\,0.1\,°C$

Table 6.1: Some of the specifications of the Moravian G2-8300 CCD camera as listed in the handbook [62]

6.1.2 Canon 5D Mark II

Additional experiments were conducted with a Canon 5D Mark II. This camera is a full-frame CMOS camera system with 21.1 megapixels. The imaging area is 36×24 *mm* with 3753×5634 pixels. ADU conversion is adjustable with ISO settings in the range $[100, 6400]$.

There are a couple of caveats when working with this camera:

- This camera features a Bayer pattern on the image sensor for color imaging. This pattern acts as wavelength filters on individual pixels. The pattern is a two-by-two matrix of the form *Green – Red – Blue – Green* going from top-to-bottom and left-to-right. To get full color information on all pixels, some sort of pixel-wise interpolation has to be applied.

  As a consequence, patch sizes in the denoising procedure must be a multiple of two, such that the Bayer pattern on patches are consistent over time.

- This camera has also been specially modified for the application in astronomical settings by the company CentralDS. It features a TEC & skiving fin heatsink cooling system which can manually be turned on or off. However, it is not possible to fix the temperature at a certain level as is the case with the Moravian camera. Therefore, chip temperature tends to increase over the duration of an exposure.

- Since it is a consumer camera, mainly used for artistic rather than scientific purposes, exact specifications are not available from the manufacturer. Moreover, the exact processing steps that are done on chip are not documented and can only be inferred from dark frame recordings.

6.2 Evaluation of Camera Calibration Methods

In this section, different calibration techniques that have been mentioned in Section 3.3.4 are evaluated and discussed.

For both cameras, a sequence of dark frames was recorded for various settings. Of each sequence, 5 images were taken aside as evaluation set and the rest were taken as a training set. Each evaluation image was processed with 4 different calibration methods. Performance was measured in PSNR, i.e., for an evaluation image I and a calibration image $\hat{I}$

$$\mathrm{PSNR}(I, \hat{I}) = 10 \cdot \log_{10}\left(\frac{I_{\mathrm{MAX}}^2}{\frac{1}{N}\sum_{i=1}^{N}(I_i - \hat{I}_i)^2}\right) \mathrm{dB} \tag{6.1}$$

was computed, with higher values corresponding to better calibration performance. One might argue that this is not a realistic performance evaluation, since it does not take into account the effects of light interacting with the sensor. However, this can also be seen as an advantage: light recordings will introduce an additional source of randomness which should not be considered for measuring performance on dark current noise alone. In this sense, this method, although less realistic, is a more rigorous evaluation.

The following 4 calibration methods were applied:

Subsequent Dark Frame Removal: The image following the evaluation dark frame in the sequence is subtracted from the evaluation image. This corresponds to the common practice of recording a dark frame immediately after capturing an image.

Pre-recorded Dark Frame Removal: The image prior to the evaluation dark frame in the sequence is subtracted from the evaluation image.

Mean Dark Frame Removal: In this setting, the mean dark frame of all recorded images in a training set for a given setting is computed and subtracted from the evaluation dark frame.

Convex Combination Dark Frame Removal: This method is described in [39]. The basic assumption of this method is that the dark frame noise present in a given recorded image can be described as the convex combination from a set of recorded dark frames, i.e., if D is the current dark frame and $\{\hat{D}^{(s)} \mid s = 1, \dots, S\}$ is the set of recorded training dark frames, than there exist weights $\alpha_s \geq 0$ such that

$$D = \sum_{s \in S} \alpha_s \hat{D}^{(s)} \tag{6.2}$$

The problem is to find the exact mixing weights α_s for a recorded light frame, since in this case D is not known. In [39], the authors propose that astronomical images are mostly smooth. Therefore, for many pixels the discrete gradient of neighboring pixels should be zero, if there is no dark frame noise present. Consequently, the α_s can be computed by solving the following quadratic optimization problem:

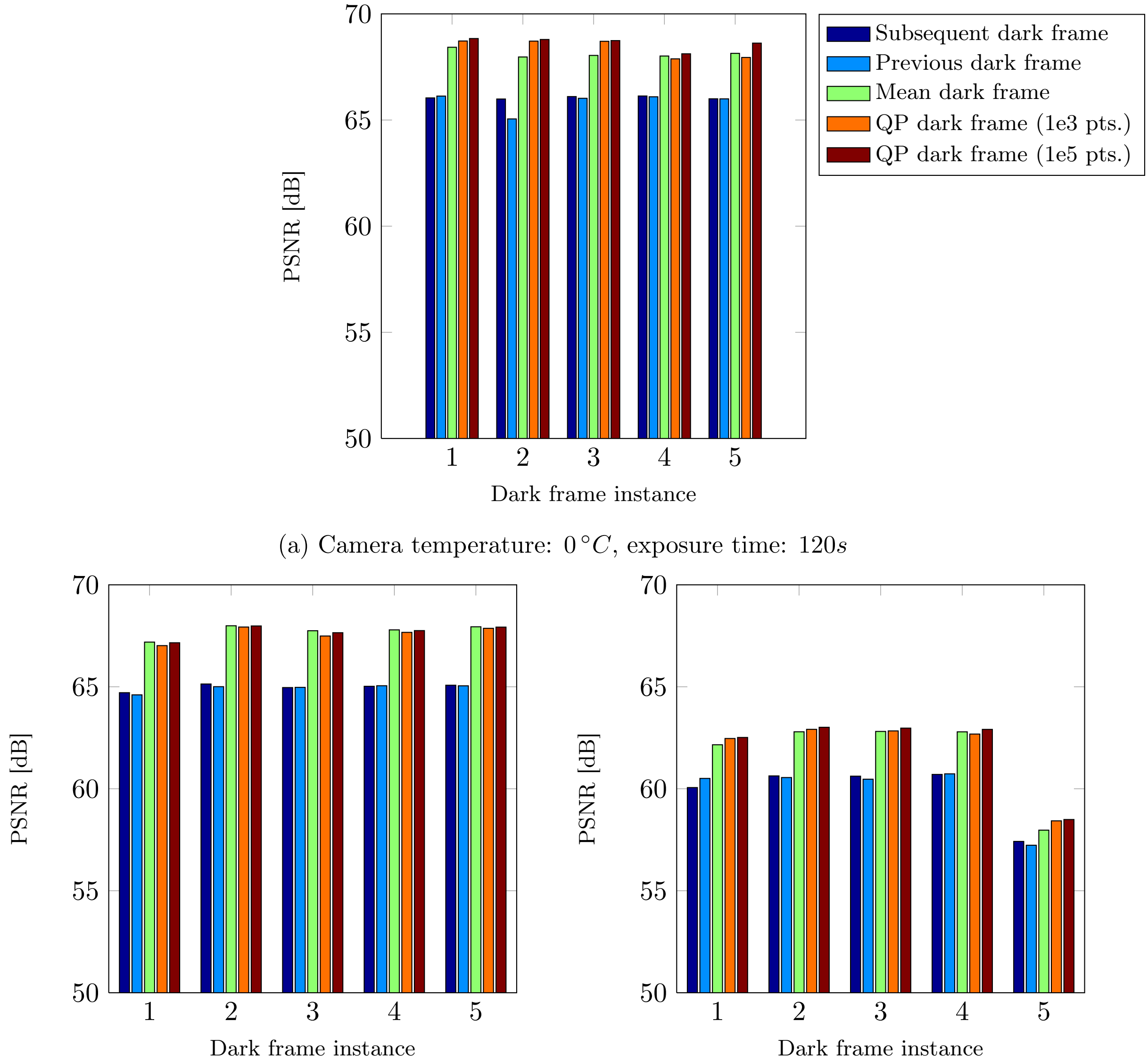
$$\begin{aligned} \min \quad & \sum_{i \in E} \sum_{j \in N_E} ((I_i - \sum_{s \in S} \alpha_s \hat{D}_i^{(s)}) - (I_j - \sum_{s \in S} \alpha_s \hat{D}_j^{(s)}))^2 && (6.3) \\ \text{s.t.} \quad & \sum_{s \in S} \alpha_s = 1 && (6.4) \\ & \alpha_s \geq 0 \qquad \forall s \in S && (6.5) \end{aligned}$$

Method	Advantages	Disadvantages
Subsequent Removal	Captures current state of the image sensor, like the sensor temperature, easy to implement both in hardware or software	Poor estimation due to small sample size, may take a lot of additional time, specially during an extended recording session
Mean Removal	Easy to implement, reduced time consumption for recording, if a lot of light frames are captured	Might not take into account current state of sensor
QP Removal [39]	Combines flexibility of adaptive dark frame estimation with improvements from taking multiple dark frames, includes other methods as special solutions	Might yield suboptimal solutions for small number of evaluation points or if smoothness assumption does not hold, computationally expensive

Table 6.2: Advantages and disadvantages of various camera calibration methods

where E is a set of evaluation points and N_E is the set of its eight neighboring pixels. All previous methods can be represented as special choices of α_s in (6.2). The here present method of evaluation gives a clear advantage to this method as compared to a realistic setting, since in this formulation $I = 0$ and therefore the smoothness assumption holds perfectly, which will not be the case for realistic images. On the other hand, this method is naturally limited by the number of evaluation points $|E|$. Here, two different settings where tested with $|E| = 10^3$ and $|E| = 10^5$ respectively.

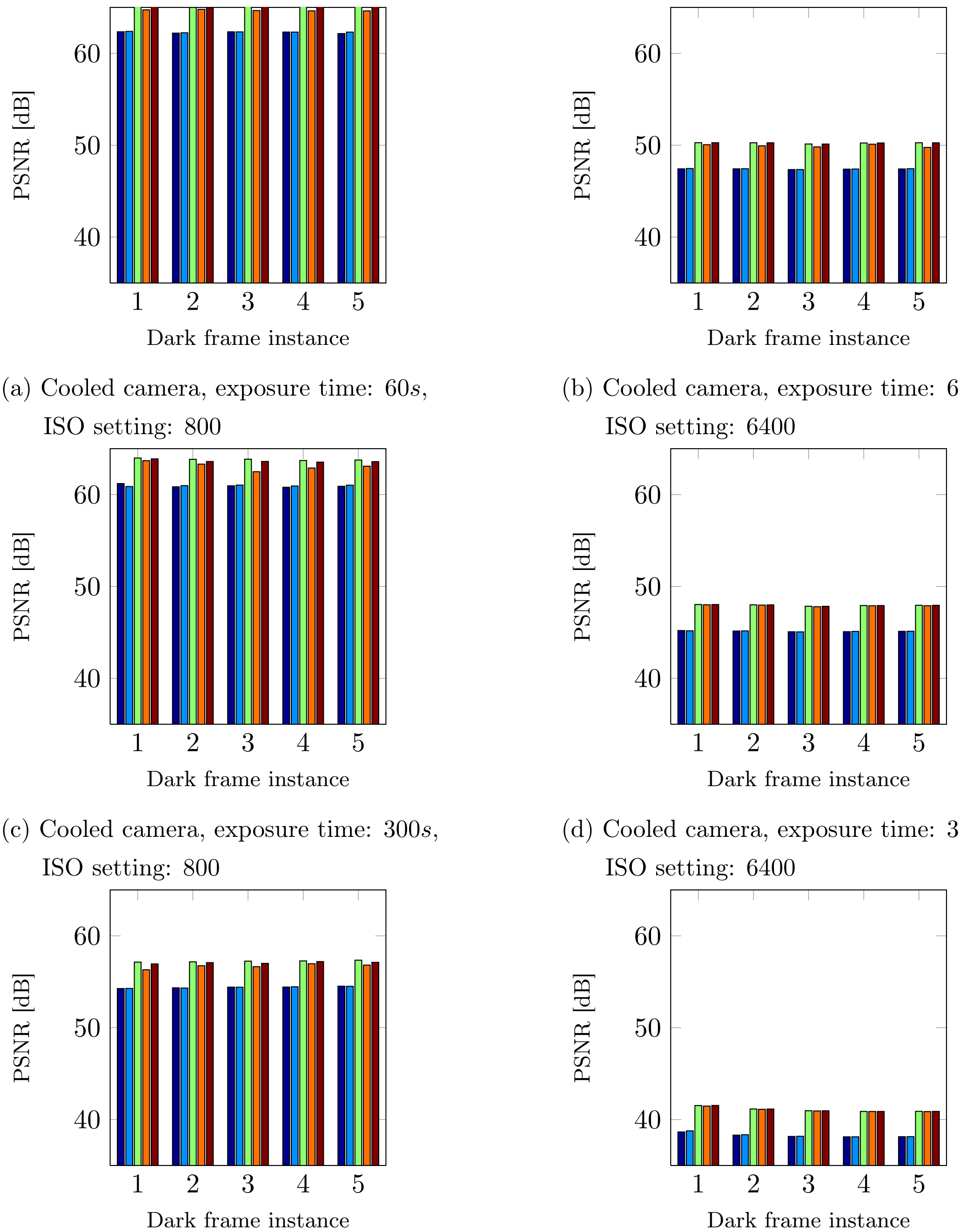
Table 6.2 lists advantages and disadvantages for each method. Figures 6.1 and 6.2 show the results for the various methods for the Moravian and Canon camera, respectively. The figure shows that combining the information from multiple dark frames always outperforms single-shot dark frame removal. Furthermore, denoising results consistently diminish with longer exposure times, higher temperatures or higher ISO settings. There are some cases, in which the solution of 6.3 is worse than the mean dark frame, even in the case of $|E| = 10^5$. Overall, the methods perform very similar. Considering that the mean is computationally much more efficient to



(a) Camera temperature: $0\,^{\circ}C$, exposure time: $120s$

(b) Camera temperature: $0\,^{\circ}C$, exposure time: $600s$ (c) Camera temperature: $18\,^{\circ}C$, exposure time: $600s$

Figure 6.1: Evaluation results of various camera calibration methods applied to Moravian dark frames. For each setting, a set of 50 dark frames was recorded and five random images were taken aside as evaluation set. Figures show PSNR in dB in comparison to the correction dark frame of the applied method. See text for detailed discussion.



(a) Cooled camera, exposure time: 60s, ISO setting: 800

(b) Cooled camera, exposure time: 60s, ISO setting: 6400

(c) Cooled camera, exposure time: 300s, ISO setting: 800

(d) Cooled camera, exposure time: 300s, ISO setting: 6400

(e) Uncooled camera, exposure time: 300s, ISO setting: 800

(f) Uncooled camera, exposure time: 300s, ISO setting: 6400

Figure 6.2: Evaluation results of various camera calibration methods applied to Canon dark frames. For each setting, a set of 30 dark frames was recorded and five random images were taken aside as evaluation set. Figures show PSNR in dB in comparison to the correction dark frame of the applied method. See text for detailed discussion.

compute, mean dark frame was used to calibrate images rather than the QP-formulation of [39].

However, comparing with related work on image denoising, it can be seen that the level of residual noise is comparatively low. If a standard signal range of $[0, 255]$ is assumed, these PSNR values of residual noise roughly correspond to AWG noise with $\sigma \in [10^{-4}, 10^{-3}]$, while common values considered in literature are $\sigma \in \{1, 5, 10, 25, 50\}$. Consequently, any algorithm to deal with residual camera noise must be able to handle very low amounts of noise.

6.3 Analysis of MLP Training Methods

Although multi-layer perceptrons are able to approximate any function arbitrarily good in general, achieving good results can be difficult [53]. Therefore, it is necessary to perform an analysis of training success for different methods and hyper-parameter settings, such as the number of hidden layers or input and output patch sizes.

In principle, a full grid-search over all combinations of considered hyper-parameters would be necessary, since it is not guaranteed that parameters can be optimized independently. However, this is not feasible in practice, since training of multi-layer perceptrons is very time consuming, often taking weeks to fully converge. Consequently, an initial analysis of training methods is needed to prune down the search tree of possible hyper-parameters.

6.3.1 Training MLPs in Low-Noise Settings

As has been shown in the previous section, removing the mean dark frame drastically reduces the amount of noise present in a captured image. Consequently, the multi-layer perceptron has to be able to deal with very low amounts of noise. However, in [21, 53] it was shown that MLPs are less effective on low noise levels. Therefore, it first has to be investigated how MLPs can be applied to the problem of removing faint residual noise.

As has been suggested in Section 5.3.3, better results might be expected with learning the residual instead of learning the actual signal. To evaluate this hypothesis and to test whether this low amount of noise can be learned at all with MLPs, artificial white Gaussian noise with small levels of σ has been added to both natural and astronomical images. MLPs were trained both on the signal and the residual, and results were compared with the state-of-the-art algorithm BM3D on the various settings.

Results can be found in Figure 6.3. The figure shows that MLPs trained on the prediction of the residual noise clearly outperform the corresponding MLP directly predicting the clean image patch. Furthermore, it can be seen that for low noise levels and in the case of signal predicting MLPs it is hard to find parameters which improve the input signal, while the initialization for residual predicting MLPs already finds values close to a local optimum.

Specially for the case of astronomical images, MLPs do not seem to improve at all with training. It is likely that the sparse distribution on the astronomical images both does not work with the usual initialization as well as causing highly irregular gradients depending on the content of the input patch, which will mostly be plain background, but every once in a while feature a star and therefore drastically alter search direction.

Consequently, in the following experiments only residual prediction was applied.

6.3.2 Evaluation of Gradient Descent Variants

To shorten overall training time and enable making more experiments, different variations of gradient descent have been tested. In these experiments, only the training performance on astronomical images with real camera noise is evaluated.

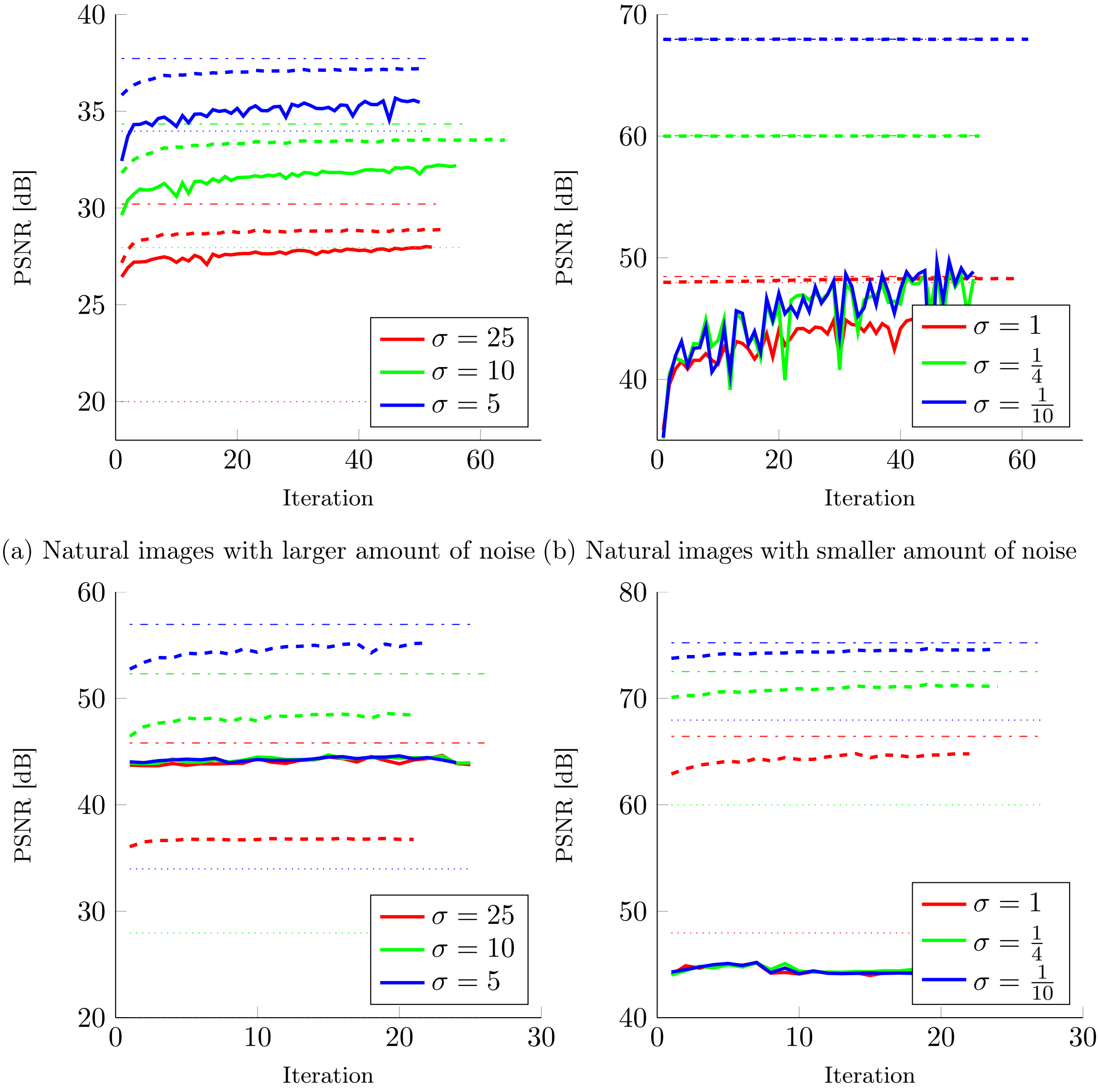
All networks presented have input patch size of 8 and output patch size of 2. MLPs have 2 hidden layers of dimension 2048 and tanh-nonlinearity. In each iteration 2^{17} patches are generated. Most experiments train on batches of 128 patch-gradients, with exception of one experiment that is done with pure stochastic gradient descent. Note that in the MLP implementation at hand, batch gradients are summed. Consequently, learning rates are multiplied with a factor of 10^{-2} to compensate for the batch size.

Figures 6.4 and 6.5 show MLP performance for different gradient descent variants. In principle, all training methods are able to improve the evaluation performance. ADAGRAD and ADADELTA both outperform stochastic gradient descent with and without momentum term. In their best configurations, ADAGRAD and ADADELTA perform similarly. ADADELTA reaches a local optimum faster but also degenerates after longer training, while ADAGRAD seems to stabilize better. SGD with single gradients finds a local optimum very quickly, but does not generalize over time.

In later experiments, ADAGRAD with learning parameter $\lambda = 10^{-3}$ and batch size of 128 was used, since it performs well both with Moravian and Canon dark frames and also is more stable.

6.3.3 Comparison of MLP Architectures for Learning Camera Noise

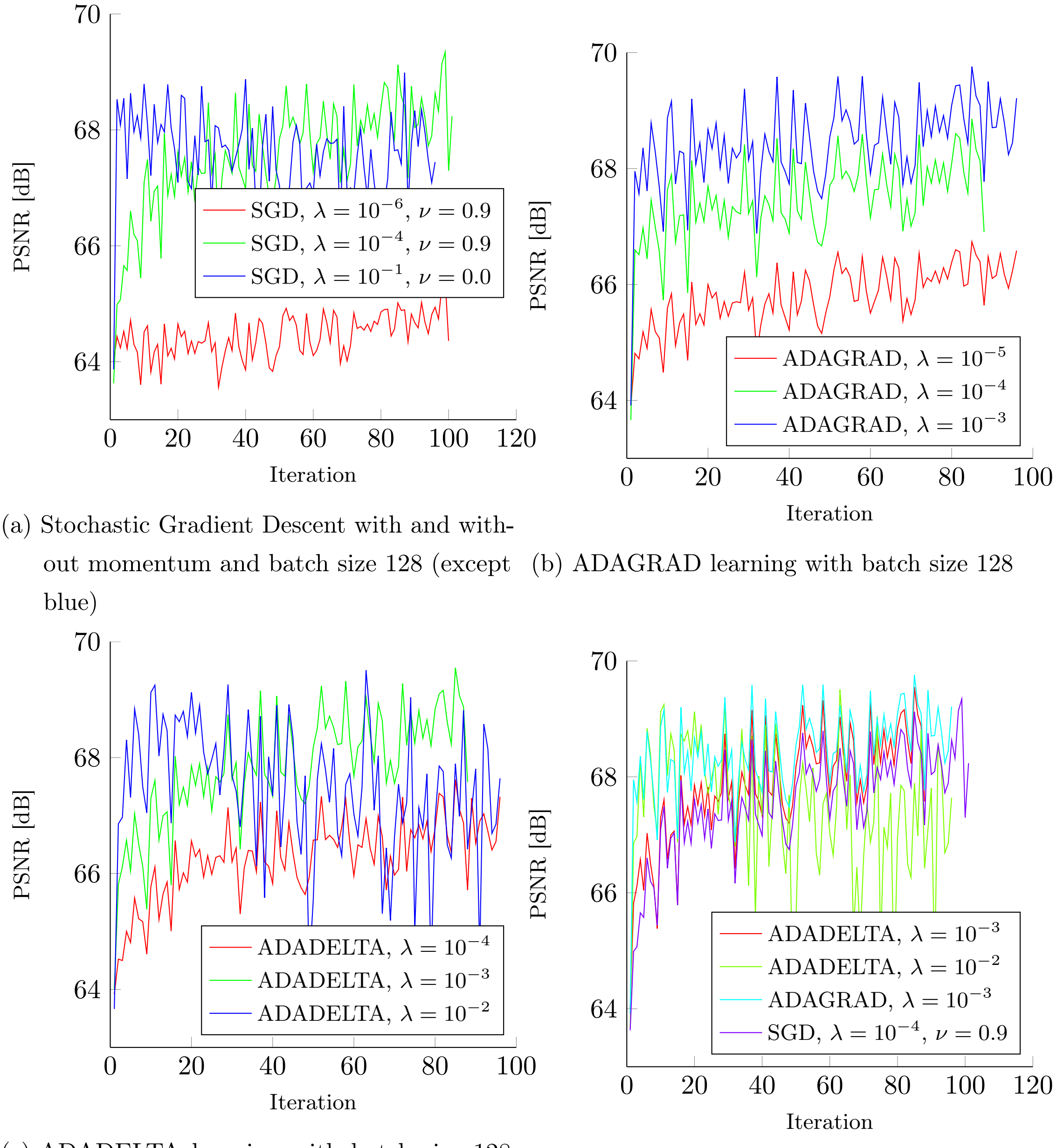
Finally, different MLP architectures were evaluated based on their performance on real camera noise.



(a) Natural images with larger amount of noise (b) Natural images with smaller amount of noise

(c) Astro-images with larger amount of noise (d) Astro-images with smaller amount of noise

Figure 6.3: Denoising performance of signal and noise precdiction MLPs for different image data sets and noise levels. Solid lines show results for signal prediction MLPs, dashed lines show results for noise prediction MLPs. Dotted lines show average PSNR of noisy input images, dash-dotted lines show average result of BM3D for comparison. Apart from noise levels and input images, exact same data was used to train the different MLPs.



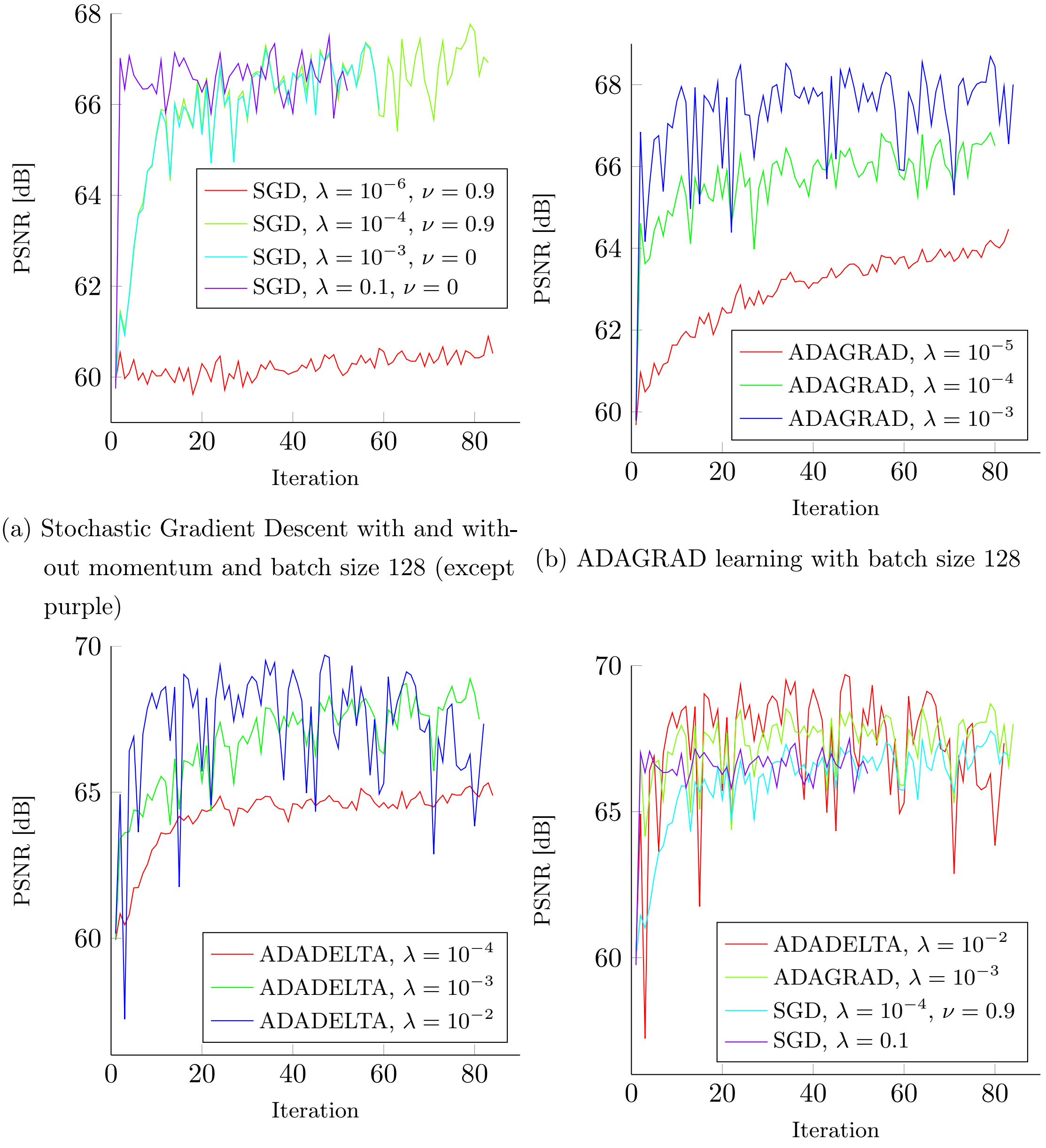
(a) Stochastic Gradient Descent with and without momentum and batch size 128 (except blue)

(b) ADAGRAD learning with batch size 128

(c) ADADELTA learning with batch size 128 and decay parameter $\delta = 0.95$

(d) Best variants of previous panels

Figure 6.4: Comparison of different gradient descent algorithms on Moravian camera noise. MLPs of identical architecture are trained with exact same training data but different gradient descent variants. Input patch size is 8 pixels, output patch size is 2 pixels. Per training iteration 2^{17} patches are generated. MLPs have two hidden layers of dimension 2048, connected with tanh-nonlinear layers. Graphs show average PSNR in dB on test image per training iteration.



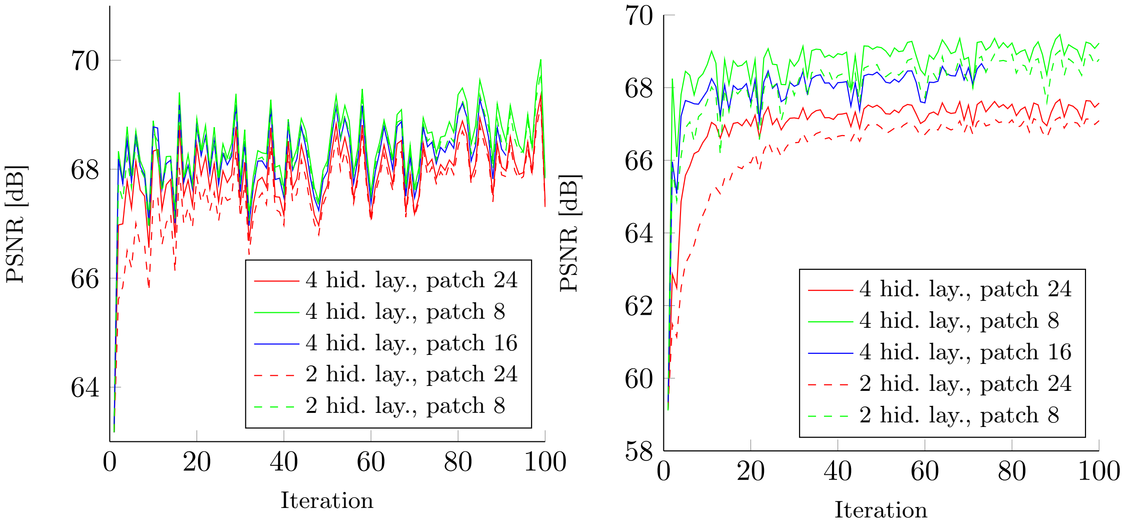
(a) Stochastic Gradient Descent with and without momentum and batch size 128 (except purple)

(b) ADAGRAD learning with batch size 128

(c) ADADELTA learning with batch size 128 and decay parameter $\delta = 0.95$

(d) Best variants of previous panels

Figure 6.5: Comparison of different gradient descent algorithms on Canon camera noise. MLPs of identical architecture are trained with exact same training data but different gradient descent variants. Input patch size is 8 pixels, output patch size is 2 pixels. Per training iteration 2^{17} patches are generated. MLPs have two hidden layers of dimension 2048, connected with tanh-nonlinear layers. Graphs show average PSNR in dB on test image set per training iteration.



(a) Moravian camera noise

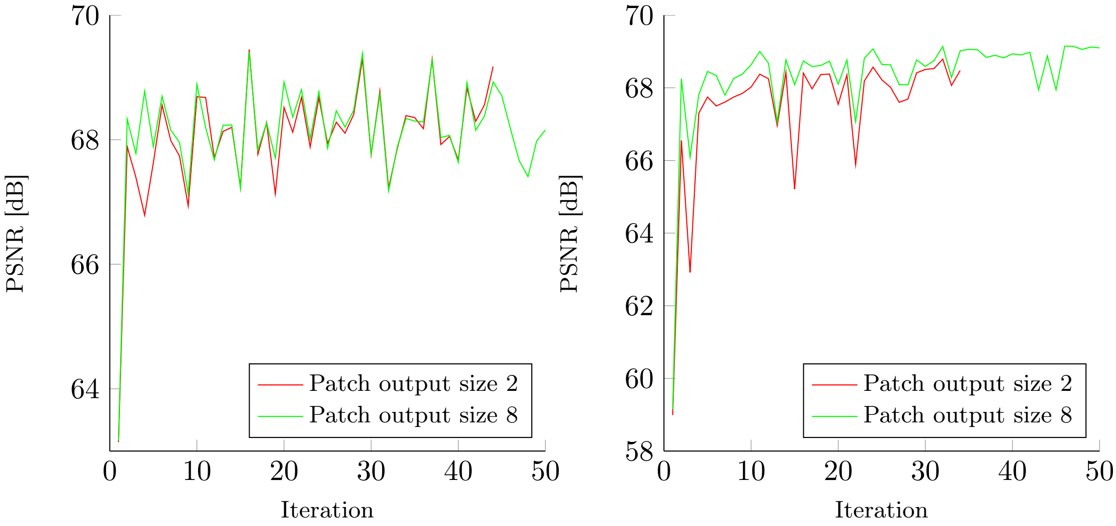
(b) Canon camera noise

Figure 6.6: Comparison of various MLP architectures for camera denoising performance. MLPs are trained with ADAGRAD with learning parameter $\lambda = 10^{-3}$ and batch size 128. Exact same training data is presented to all MLPs for each type of noise. MLPs differ in the number of hidden layers and patch sizes. Input and output sizes are kept identical for this comparison. Graphs show average PSNR in dB on test image set per training iteration.

First, it was tested how the number of hidden layers and patch input/output sizes affect the evaluation performance of the camera denoising task. As before, MLPs are trained with the ADAGRAD gradient descent algorithm with batch size of 128. In each iteration, 2^{17} training patches were generated. For all these tests, the input patch size is equal to the output patch size.

The results can be seen in Figure 6.6. For both cameras, 4 hidden layers outperform 2 hidden layers consistently. Therefore, it can be concluded that no overfitting occurs and learning the camera noise is a difficult problem even with sparse image distributions. Secondly, smaller patch sizes outperform larger patches sizes, again for both cameras. One possible explanation for this observation is that the image context is not as important in astronomical images as it is in natural images.

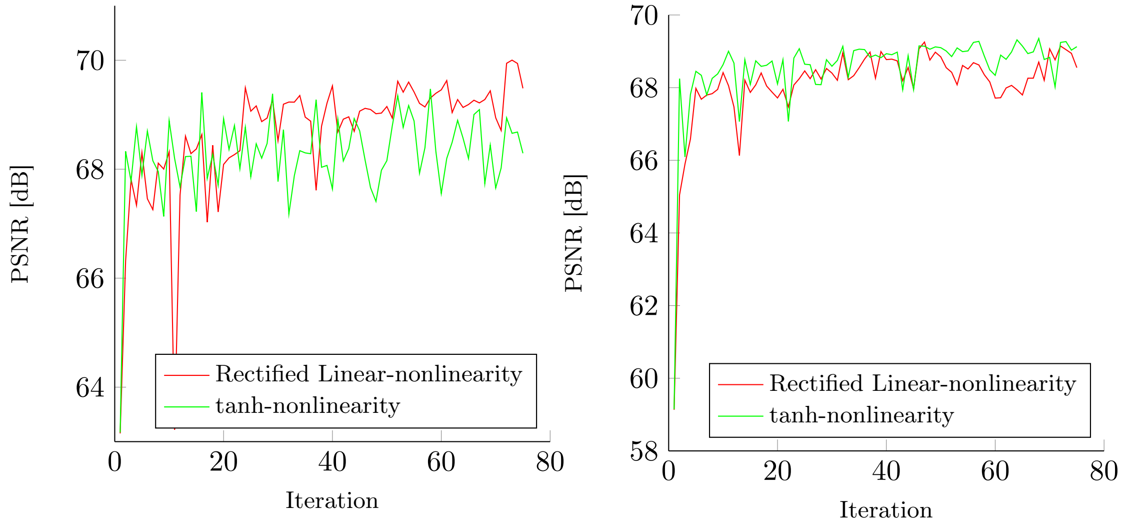
As second question concerns the optimal size of the output patches relative to the size of input patches. A decrease in size from input to output patch leads to a reduced probability space which has to be learned accurately, while choosing identical sizes in input and output patch leads to



(a) Moravian Camera Noise

(b) Canon Camera Noise

Figure 6.7: Comparison of various MLP architectures for camera denoising performance. MLPs are trained with ADAGRAD with learning parameter $\lambda = 10^{-3}$ and batch size 128. Exact same training data is presented to all MLPs for each type of noise. MLPs have two hidden layers of dimension 2048 and use tanh-nonlinear layers. differ only in output patch size. Graphs show average PSNR in dB on test image set per training iteration.



(a) Moravian Camera Noise

(b) Canon Camera Noise

Figure 6.8: Comparison of various nonlinearities for camera denoising performance. MLPs are trained with ADAGRAD with learning parameter $\lambda = 10^{-3}$ and batch size 128. Exact same training data is presented to all MLPs for each type of noise. MLPs have two hidden layers of dimension 2048 and input/output patch size of 8 pixels. MLPs differ in the choice of nonlinear layers connecting the linear layers. One net uses tanh-layers, the other uses rectified linear layers. See Table 5.1 in Section 5.3.2 for details. Graphs show average PSNR in dB on test image set per training iteration.

overlap of predictions that have to be averaged, which may or may not lead to improvements in prediction. To this end, two MLPs are trained: one has output patch size same as input patch size and one has output patch size of 2, the smallest possible amount output patch size taking into account the Bayer pattern on the Canon camera. To increase training speed, the sliding window stride in the second case was increased from 2 to 4, giving to predictions per pixel.

As Figure 6.7 shows, having smaller output than input sizes does not help with astronomical images. Also, it was found that evaluation performance increased consistently by further decreasing the patch offset.

Finally, for the best architectures, the usage of different nonlinearities were compared. Here, the outcomes differ for the two different cameras, as can be seen in Figure 6.8. While both the tanh- and the rectified linear-nonlinearities are generally able to achieve good evaluation performance, the rectified linear units were found to work better on Moravian dark frames, whereas tanh-layers work better on Canon dark frames.

6.4 Evaluation of Training Data for Camera-specific Image Denoising

It is known that the training data can have significant impact on the evaluation performance of the MLP. As has been illustrated in Figure 1.1, inspecting the problem formulation, this thesis adjusts the basic model in two general categories:

1. Considering camera-specific residual noise instead of AWG noise

2. Denoising astronomical images instead of natural images

Instead of interchanging both data sources at once, it should also be tested how each individual step will affect denoising performance.

Furthermore, two astronomical image data sets with different properties have been collected and are available.

This part will evaluate which effect the different sources of training data sets will have on the denoising performance.

6.4.1 Evaluation of Image Data Sets

First, the influence of the image data sets was evaluated. To this end, MLPs of identical structure are trained on camera noise and images from the ImageNet, the Capella, and the SDSS data set, respectively. MLPs have input patch size of 8 and output patch size of 2, with two additional hidden layers of dimension 2048, connected with tanh-layers. All are trained with the ADAGRAD algorithm with learning rate $\lambda = 10^{-3}$ and batch size of 128.

Tables 6.3 and 6.4 show the average denoising result in PSNR dB for each combination of dark frame and evaluation image in the specific set. The results clearly show that denoising performance is best, if training and test images come from the same data set, as has been expected. Interestingly, MLPs trained on the SDSS data perform better on natural images than MLPs trained on the Capella data set. This is probably due to the fact that there are only very few images in the Capella data set and generalization is therefore not so good.

Evaluation set	**ImageNet**	**Capella**	**SDSS**
Training set			
ImageNet	**47.1**	57.9	59.1
Capella	46.9	**59.9**	63.3
SDSS	47.0	55.0	**68.5**

Table 6.3: Comparison of different imaga data sets for removing Moravian noise. MLPs of identical configuration are trained until convergence for each category of training data as evaluated on test images from the same category. Training data is generated as presented in Section 5.1.1 with noise free images from the respective image data set and dark frames from the Moravian camera. Dark frames are exposed 600 seconds while the camera temperature is regulated to a setpoint of $18\,{}^{\circ}C$. Reported are average denoising results on respective test images in dB PSNR. Best results for each evaluation set are shown **bold**.

Evaluation set	**ImageNet**	**Capella**	**SDSS**
Training set			
ImageNet	**54.9**	57.1	63.6
Capella	52.4	**57.2**	59.4
SDSS	53.9	52.9	**68.2**

Table 6.4: Comparison of different imaga data sets for removing Canon noise. MLPs of identical configuration are trained until convergence for each category of training data as evaluated on test images from the same category. Training data is generated as presented in Section 5.1.1 with noise free images from the respective image data set and dark frames from the Canon camera. Dark frames are exposed 300 seconds with an ISO setting of 800. The camera was not cooled during exposure. Reported are average denoising results on respective test images in dB PSNR. Best results for each evaluation set are shown **bold**.

		Natural images		**Astro-images**	
	Evaluation set	**AWGN**	**Camera**	**AWGN**	**Camera**
		$\sigma = 6.07 \cdot 10^{-4}$	**noise**	$\sigma = 6.07 \cdot 10^{-4}$	**noise**
	Training set				
Natural images	**AWGN** $\sigma = 6.07 \cdot 10^{-4}$	**64.3**	47.0	65.2	63.4
	Camera noise	61.4	**47.1**	65.5	59.1
Astro-images	**AWGN** $\sigma = 6.07 \cdot 10^{-4}$	62.4	47.0	**72.8**	**68.6**
	Camera noise	61.1	47.0	70.7	68.5

Table 6.5: Comparison of different training data for removing Moravian noise. MLPs of identical configuration are trained until convergence for each category of training data as evaluated on test images from the same category. MLPs are then evaluated on images from the respective other categories. Reported are average denoising results on respective test images in dB PSNR. Best results for each evaluation set are shown **bold**.

6.4.2 Evaluation of Noise Data

Finally, it was tested how the noise source in the training images affects the test performance. Four additional MLPs were trained on natural and SDSS images respectively with additive white Gaussian noise on comparable level to each camera. For each camera, the standard deviation of each pixel was computed on the training dark frames and the mean over all pixel standard deviations was used as noise level σ. Noise levels of $\sigma = 6.07 \cdot 10^{-4}$ and $\sigma = 1.3 \cdot 10^{-3}$ were found for the Moravian and Canon dark frames respectively.

The comparison for the different configurations can be found in Table 6.5 and Table 6.6 for the Moravian and Canon camera respectively. Interestingly, it is found that training on artificial additive white Gaussian noise works better than to train on data consisting of actual residual noise.

There are multiple possible explanation for these results. For instance, there may not have

		Natural images		**Astro-images**	
	Evaluation set	**AWGN**	**Camera**	**AWGN**	**Camera**
		$\sigma = 1.3 \cdot 10^{-3}$	**noise**	$\sigma = 1.3 \cdot 10^{-3}$	**noise**
	Training set				
Natural images	**AWGN** $\sigma = 1.3 \cdot 10^{-3}$	**57.7**	**54.9**	59.6	59.4
	Camera noise	55.4	**54.9**	59.3	63.6
Astro-images	**AWGN** $\sigma = 1.3 \cdot 10^{-3}$	56.4	54.1	**70.0**	**69.4**
	Camera noise	55.9	53.9	68.6	68.2

Table 6.6: Comparison of different training data for removing Canon noise. MLPs of identical configuration are trained until convergence for each category of training data as evaluated on test images from the same category. MLPs are then evaluated on images from the respective other categories. Reported are average denoising results on respective test images in dB PSNR. Best results for each evaluation set are shown **bold**.

been enough dark frames in the training set, such that the number of patches for each patch location was not sufficient enough. Furthermore, since the MLPs are trained on patches, it could very well be that a pattern that might be apparent on image level is lost if only small patches are considered, disappears on patch-level, i.e., residual noise is statistically indistinguishable from additive white Gaussian noise on patch-level, even though there might be a pattern on global image scale.

6.5 Comparison of MLPs with Existing Methods

To evaluate the overall performance of the multi-layer perceptron for learning camera-specific image denoising, the MLPs with the best test performance during training are compared against state-of-the-art denoising algorithms in camera-specific in AWG noise respectively. In the case of AWGN, BM3D [1] is applied. For camera-specific denoising, the DF-MAP method of Burger et al. [37] is used. Note that the DF-Mean algorithm which is proposed in [37] corresponds to the noisy input image in this case, i.e., DF-Mean is the necessary calibration step which will be performed prior to applying any other method.

To enable a fair comparison, a full parameter search is done for DF-MAP, whereby it was found that in this setting the parameter choice $p = 1$ and $\lambda = 10^{-2}$ works best. Since BM3D assumes a known σ, two different cases are considered:

1. Similar to the experiments in Section 6.4.2, the mean of the pixel-wise standard deviation measured on the training dark frames is considered to be the unknown σ.

2. For each generated test image, the true σ as is given by the noise is computed and passed to BM3D, i.e.,

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}((y_i - x_i) - (\overline{y_i - x_i}))^2} \tag{6.6}$$

   This choice of σ, though theoretically perfect, is very unrealistic in practical settings, since it can be seen in Equation (6.6) that the unknown true image x needs to be known to compute the exact σ. Therefore, this represents a best case scenario for BM3D.

In practice, it is expected that the real denoising performance of BM3D lies somewhere between this lower and upper bound.

Tables 6.7 and 6.8 show the average denoising performance for all dark frames in the test set of each algorithm and each test image. In the case of the Moravian dark frames, the MLPs clearly

	Test Image 1			Test Image 2			Test Image 3		
Filter	*g*	*i*	*r*	*g*	*i*	*r*	*g*	*i*	*r*
Method									
DF-Mean (Noisy Input)	62.2	62.2	62.2	62.2	62.2	62.2	62.2	62.2	62.2
BM3D (unknown σ)	66.0	65.3	65.5	64.7	64.8	65.1	65.4	64.3	65.3
BM3D (known σ)	69.3	68.4	68.8	67.4	67.5	68.3	68.1	66.6	68.1
DF-MAP ($p = 1$)	67.5	66.8	67.1	66.3	66.4	66.9	66.0	65.1	66.0
DF-MAP ($p = 1.4$)	67.9	66.5	66.9	67.3	67.4	68.1	65.7	64.8	65.8
DF-MAP ($p = 2$)	63.8	62.5	62.7	64.8	64.8	64.8	62.6	62.4	62.5
MLP (Trained on AWGN)	70.2	**69.0**	**69.5**	68.0	68.1	68.9	**68.3**	**66.9**	**68.3**
MLP (Trained on Camera)	**70.4**	65.2	67.1	**68.9**	**69.2**	**70.2**	65.9	64.9	66.2

Table 6.7: Comparison of different denoising algorithms on Moravian-specific noise. For each test instance, the average denoising performance over all different test dark frames are shown. Best results are shown **bold** for each image.

	Test Image 1			Test Image 2			Test Image 3		
Filter	*g*	*i*	*r*	*g*	*i*	*r*	*g*	*i*	*r*
Method									
DF-Mean (Noisy Input)	57.4	57.4	57.4	57.4	57.4	57.4	57.4	57.4	57.4
BM3D (unknown σ)	60.7	60.0	60.2	58.9	58.9	59.6	60.2	58.8	60.1
BM3D (known σ)	**74.0**	**71.2**	**72.1**	**70.3**	**70.6**	**72.1**	**69.0**	**66.8**	**69.1**
DF-MAP ($p = 1$)	72.0	70.2	70.8	69.3	69.5	70.7	66.9	65.1	67.1
DF-MAP ($p = 1.4$)	69.0	66.6	67.2	68.6	68.8	69.8	64.8	63.5	64.9
DF-MAP ($p = 2$)	59.3	57.2	57.9	60.3	60.2	60.3	57.6	56.0	57.2
MLP (Trained on AWGN)	73.5	70.2	71.6	69.5	69.7	71.6	67.7	65.8	68.0
MLP (Trained on Camera)	71.3	67.3	68.7	69.2	69.3	71.1	65.9	64.6	66.1

Table 6.8: Comparison of different denoising algorithms on Canon-specific noise. For each test instance, the average denoising performance over all different test dark frames are shown. Best results are shown **bold** for each image.

outperform current state-of-the-art algorithm. Furthermore it can also be seen that the MLP specifically trained on this type of noise sometimes excels while only performing mediocre in other cases. The MLP trained on additive white Gaussian noise with a comparable level of noise, however, performs very good across all images. Figures 6.9 - 6.11 show the clean test images as well as a detailed analysis of two interesting regions. In accordance of previous findings [21], it can bee seen that figure 6.11 is very smooth and of regular structure, which might explain the worse results for the MLP.

In the case of Canon dark frames, however, no improvements over existing methods could be reached. A possible explanation for this could be that the AWGN model fits good the residual noise of this camera. This is supported by the observation that the ratio of mean pixel-wise standard deviation vs. standard deviation of pixel-wise standard deviation is higher for the Canon camera. This implies that there is less pattern in the residual noise for the Canon camera than there is for the Moravian camera.

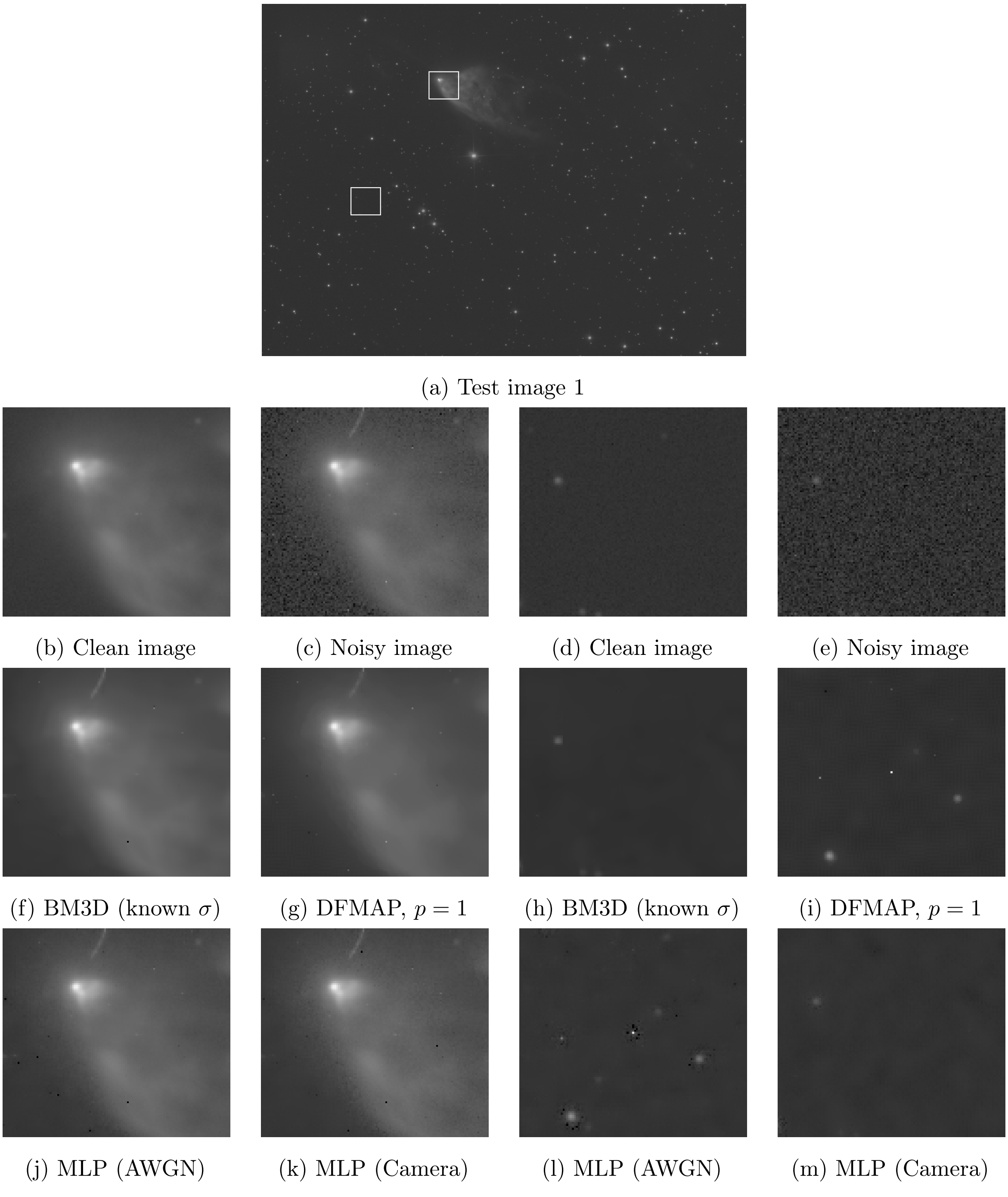
6.6 Discussion

Contrasting the results from Sections 6.2 and 6.5 with literature on AWGN image denoising, it is found that residual dark current noise after camera calibration is much less than as is currently considered in many publications. In fact, the amount of residual noise is even lower than the signal resolution in JPEG images or other 8-bit image formats. Therefore, this small amount of noise, while still being an important factor in astro-photography, cannot be analyzed in the traditional denoising research evaluation setup. By taking high-resolution scientific image data as well as high-resolution raw camera data, it is possible to extend the evaluation while keeping it comparable to the existing literature at the same time.

Additionally, it was found that it is possible to generate training and evaluation data corrupted by actual camera noise. Although this model needs yet to incorporate more noise sources, such as the photon shot noise as well as pixel-response non-uniformity, it is possible to compare results quantitatively in astronomical image denoising in practice, which has previously not been possible.

The results also show that significant denoising improvement can be achieved even on low noise levels for astronomical images. More specifically, relative improvements on astronomical images are *more than a magnitude larger* than on natural images of the same level of noisiness. This clearly indicates that image denoising in astronomical images is more than a special of natural image denoising.

At the same time, it has to be considered that both evaluation and training has been done on



(a) Test image 1

(b) Clean image

(c) Noisy image

(d) Clean image

(e) Noisy image

(f) BM3D (known σ)

(g) DFMAP, $p = 1$

(h) BM3D (known σ)

(i) DFMAP, $p = 1$

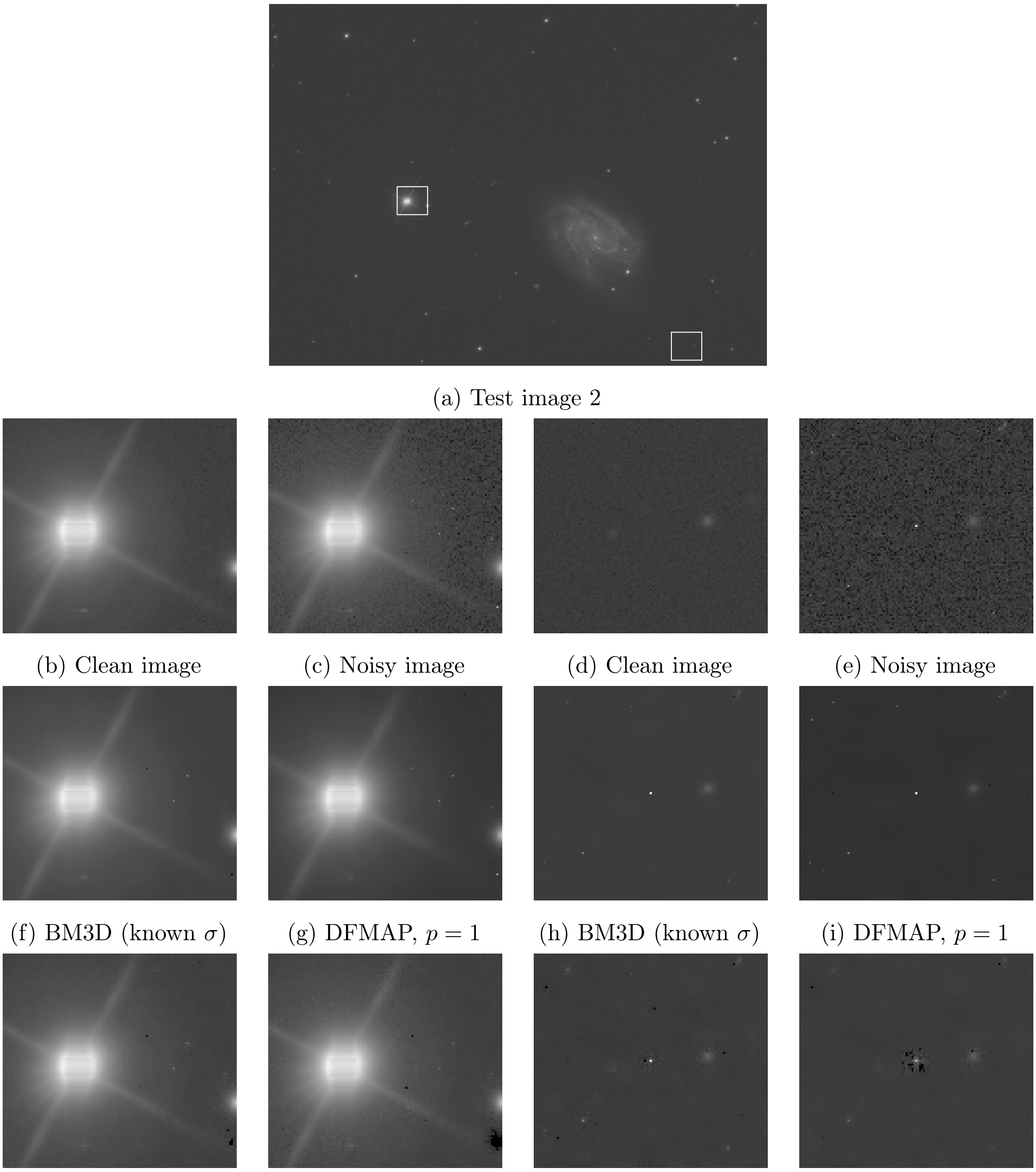
(j) MLP (AWGN)

(k) MLP (Camera)

(l) MLP (AWGN)

(m) MLP (Camera)

Figure 6.9: The main image shows the g-filter band of test image 1. White boxes indicate two regions of interest which are depicted in large below. For each region of interest, the clean version, the noise-corrupted version and results from 4 different denoising algorithms are shown.



(a) Test image 2

(b) Clean image

(c) Noisy image

(d) Clean image

(e) Noisy image

(f) BM3D (known σ)

(g) DFMAP, $p = 1$

(h) BM3D (known σ)

(i) DFMAP, $p = 1$

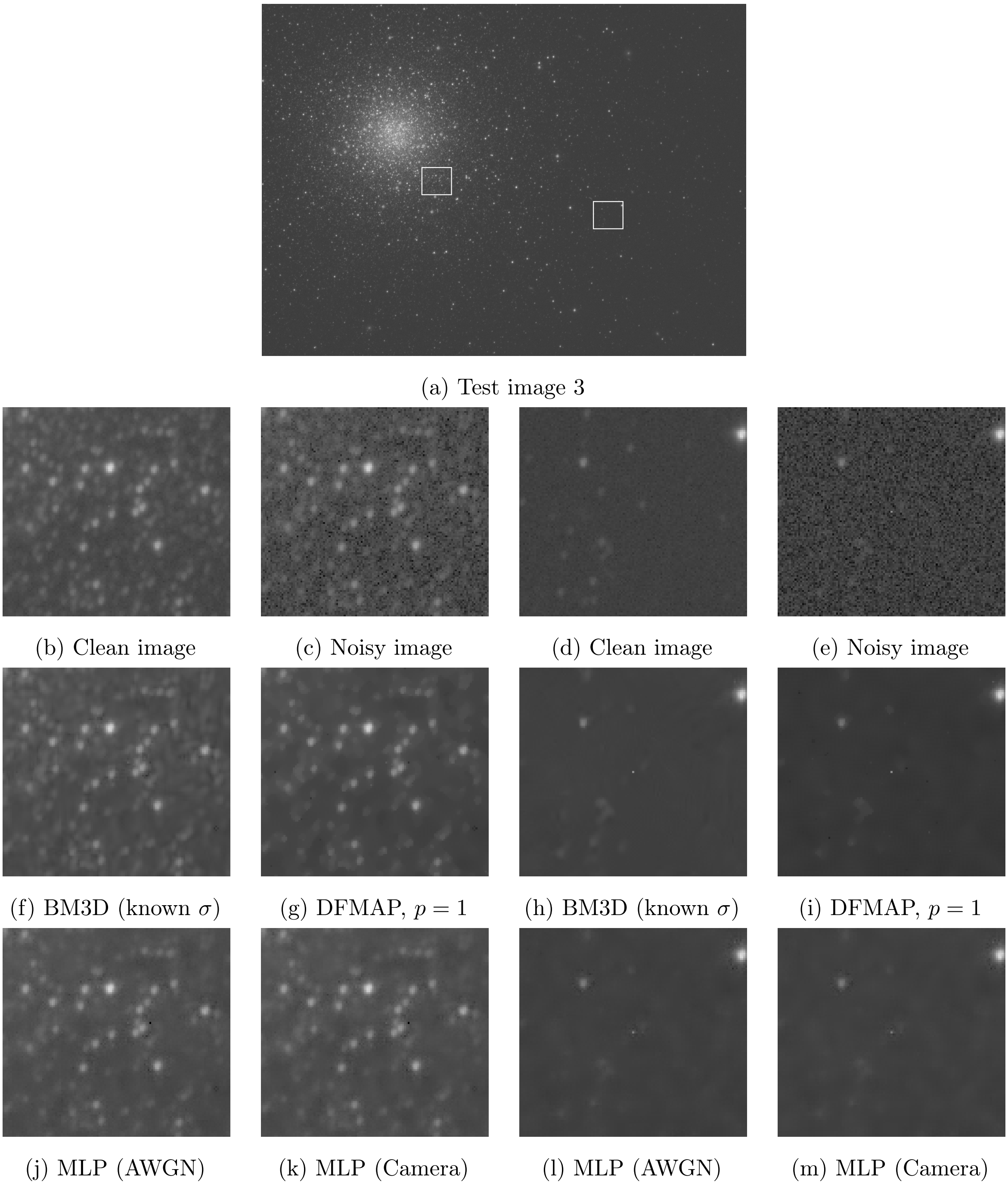
(j) MLP (AWGN)

(k) MLP (Camera)

(l) MLP (AWGN)

(m) MLP (Camera)

Figure 6.10: The main image shows the r-filter band of test image 2. White boxes indicate two regions of interest which are depicted in large below. For each region of interest, the clean version, the noise-corrupted version and results from 4 different denoising algorithms are shown.



(a) Test image 3

(b) Clean image

(c) Noisy image

(d) Clean image

(e) Noisy image

(f) BM3D (known σ)

(g) DFMAP, $p = 1$

(h) BM3D (known σ)

(i) DFMAP, $p = 1$

(j) MLP (AWGN)

(k) MLP (Camera)

(l) MLP (AWGN)

(m) MLP (Camera)

Figure 6.11: The main image shows the i-filter band of test image 3. White boxes indicate two regions of interest which are depicted in large below. For each region of interest, the clean version, the noise-corrupted version and results from 4 different denoising algorithms are shown.

„noisy“ astronomical image data: even though the images from the Sloan Digital Sky Survey are not degraded by measurable camera noise, there is still inherent noise from background sky flux. There are two implications of this problem:

1. Since evaluation is also done with inherent noisy images, performance is measured on *how accurately the denoising algorithm can reproduce an image with background sky flux noise.* Furthermore, training of machine learning algorithms for astronomical image denoising could also possibly improved by having noise free images, since currently the sky flux noise has to be learned, it has to be distinguished from camera noise and it has to be restored in order to perform well.

2. While it will never be possible to physically measure a noise free astronomical image, it might be possible to fully generate artificial but realistic astronomical images, i.e., to *find a generative model for astronomical images.* It is an open research question how a generative model for astronomical images could look like and how training samples from this hypothetical generative model could influence denoising results.

Considering the application of MLPs for image denoising, it has been shown that it is both possible and even beneficial to learn the residual noise pattern on an image patch instead of predicting the clean patch directly. In the case of astronomical image patches this has been the only successful way to train a neural network for image denoising.

Recently, there has been growing research interest in understanding the functioning principles of multi-layer perceptrons [63, 64, 53]. Now, the question arises *how noise is represented* in a neural network. Additionally, since a model of the signal distribution must somehow also be inherent in a trained MLP, it would be interesting to know if a direct signal model could be constructed from the noise model. Similarly, it remains an open question how MLP training has to be adopted to train directly on astro-images, since theory dictates that this should also be possible in general.

7 Conclusions

7.1 Summary

In this work, the general research problem of image denoising was specialized to incorporate more specific noise information from digital cameras. The general theory of image denoising has been discussed for this context and considerations for possible algorithms have been presented.

Secondly, an important application of camera-specific image denoising, namely denoising of digital astro-photographs, has been presented in detail and contrasted to the general case.

Furthermore, by presenting a framework for artificially generating training instances of image pairs of realistic clean and noisy astronomical images, a rigorous empirical evaluation has been made possible. This has been used to train a machine learning algorithm for camera-specific image denoising. By training a multi-layer perceptron, state-of-the-art results have been achieved. Additionally, training methods for MLPs have empirically been studied and a necessary transformation of the training problem has been identified.

7.2 Outlook

Taking the research from this point, there are several key aspects which need more detailed analysis in future work.

Calibration and evaluation protocols While this thesis presents one possible method of making denoising performance quantifiable for astronomical imaging, more work needs to be done to incorporate all kinds of image degenerations into the model while keeping evaluation feasible in practice. In this case, this was achieved by decoupling the noise capturing process from the signal generating process such that both components could be combined independently while preserving the properties of actual camera noise and having perfect information over the ground truth.

Models for astronomical images As is the case for natural images, defining a suitable image prior or other model of astronomical images is a hard and as-of-yet unsolved problem. This is an important problem, since a model of astronomical images would both allow for generating noise free training data as well as lead to the construction on more specific machine learning methods for this problem setup.

Modeling of the complete imaging pipeline In practice, astronomical images also suffer from further noise sources, both internal and external of the camera. Capturing long-exposure photographs requires guiding of the telescope which will introduce additional noise. Lens imperfections and the atmosphere distorts the light from stars further. By developing models that also take these noise sources into account, further improvements could be achieved.

Bibliography

[1] K. Dabov, A. Foi, et al. „Image denoising by sparse 3-D transform-domain collaborative filtering“. In: *IEEE Transactions on Image Processing (TIP)* 16.8 (2007), pp. 2080–2095.

[2] C. Tomasi and R. Manduchi. „Bilateral filtering for gray and color images“. In: *Proceedings of the Sixth International Conference on Computer Vision (ICCV)*. 1998, pp. 839–846.

[3] G. Gerig, O. Kubler, et al. „Nonlinear anisotropic filtering of MRI data“. In: *IEEE Transcations on Medical Imaging* 11.2 (1992), pp. 221–232.

[4] J. Portilla, V. Strela, et al. „Image denoising using scale mixtures of Gaussians in the wavelet domain“. In: *IEEE Transactions on Image Processing (TIP)* 12.11 (2003), pp. 1338–1351.

[5] M. Elad and M. Aharon. „Image denoising via sparse and redundant representations over learned dictionaries“. In: *Transactions on Image Processing (TIP)* 15.12 (2006), pp. 3736–3745.

[6] S. Roth and M. Black. „Fields of experts: A framework for learning image priors“. In: *International Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 2. IEEE. 2005, pp. 860–867.

[7] H. Burger, C. Schuler, and S. Harmeling. „Image denoising: Can plain neural networks compete with BM3D?“ In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2012), pp. 2392–2399.

[8] A. Buades, B. Coll, and J. Morel. „A non-local algorithm for image denoising“. In: *International Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 2. IEEE. 2005, pp. 60–65.

[9] M. Zontak and M. Irani. „Internal statistics of a single natural image“. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. 2011, pp. 977–984. DOI: 10.1109/CVPR.2011.5995401.

[10] L. Sun and J. Hays. „Super-resolution from internet-scale scene matching“. In: *Computational Photography (ICCP), 2012 IEEE International Conference on.* 2012, pp. 1–12. DOI: 10.1109/ICCPhot.2012.6215221.

[11] S. Lefkimmiatis, P. Maragos, and G. Papandreou. „Bayesian Inference on Multiscale Models for Poisson Intensity Estimation: Applications to Photon-Limited Image Denoising“. In: *Image Processing, IEEE Transactions on* 18.8 (2009), pp. 1724–1741. ISSN: 1057-7149. DOI: 10.1109/TIP.2009.2022008.

[12] F. Luisier, T. Blu, and M. Unser. „Image Denoising in Mixed Poisson–Gaussian Noise“. In: *IEEE Transactions on Image Processing (TIP)* 20.3 (2011), pp. 696–708.

[13] M. Mäkitalo and A. Foi. „Poisson-gaussian denoising using the exact unbiased inverse of the generalized anscombe transformation“. In: *International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE. 2012, pp. 1081–1084.

[14] D. A. F. Florencio and R. W. Schafer. *Decision-based median filter using local signal statistics.* 1994. DOI: 10.1117/12.185969. URL: http://dx.doi.org/10.1117/12.185969.

[15] T. Chen and H. R. Wu. „Adaptive impulse detection using center-weighted median filters“. In: *Signal Processing Letters, IEEE* 8.1 (2001), pp. 1–3. ISSN: 1070-9908. DOI: 10.1109/97.889633.

[16] P.-E. Ng and K.-K. Ma. „A switching median filter with boundary discriminative noise detection for extremely corrupted images“. In: *Image Processing, IEEE Transactions on* 15.6 (2006), pp. 1506–1516. ISSN: 1057-7149. DOI: 10.1109/TIP.2005.871129.

[17] R. Garnett, T. Huegerich, et al. „A universal noise removal algorithm with an impulse detector“. In: *Image Processing, IEEE Transactions on* 14.11 (2005), pp. 1747–1754. ISSN: 1057-7149. DOI: 10.1109/TIP.2005.857261.

[18] Y. Weiss and W. Freeman. „What makes a good model of natural images?“ In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2007, pp. 1–8.

[19] D. Zoran and Y. Weiss. „From learning models of natural image patches to whole image restoration“. In: *International Conference on Computer Vision (ICCV).* IEEE. 2011, pp. 479–486.

[20] V. Jain and H. Seung. „Natural image denoising with convolutional networks“. In: *Advances in Neural Information Processing Systems (NIPS)* 21 (2008), pp. 769–776.

[21] H. C. Burger, C. J. Schuler, and S. Harmeling. „Image denoising with multi-layer perceptrons, part 1: comparison with existing algorithms and with bounds“. In: *arXiv:1211.1544* (2012).

[22] C. Schuler, B. H.C., et al. „A machine learning approach for image deconvolution“. In: *Submitted to the Conference on Computer Vision and Pattern Recognition (CVPR)* (2013), pp. 1–8.

[23] H. Tian. „Noise analysis in CMOS image sensors“. PhD thesis. Stanford University, 2000.

[24] W. S. Boyle. „Charge coupled semiconductor devices“. In: *BSTJ* 49 (1970), pp. 587–593.

[25] C. D. Mackay. „Charge-coupled devices in astronomy“. In: *Annual Review of Astronomy and Astrophysics* 24 (1986), pp. 255–283. DOI: `10.1146/annurev.aa.24.090186.001351`.

[26] J. R. Janesick, T. Elliott, et al. „Scientific Charge-Coupled Devices“. In: *Optical Engineering* 26.8 (1987), pp. 268692–268692–. DOI: `10.1117/12.7974139`. URL: `http://dx.doi.org/10.1117/12.7974139`.

[27] E. R. Fossum. *Active pixel sensors: are CCDs dinosaurs?* 1993. DOI: `10.1117/12.148585`. URL: `http://dx.doi.org/10.1117/12.148585`.

[28] S. Mendis, S. Kemeny, and E. Fossum. „CMOS active pixel image sensor“. In: *Electron Devices, IEEE Transactions on* 41.3 (1994), pp. 452–453. ISSN: 0018-9383. DOI: `10.1109/16.275235`.

[29] S. Mendis, S. Kemeny, et al. „CMOS active pixel image sensors for highly integrated imaging systems“. In: *Solid-State Circuits, IEEE Journal of* 32.2 (1997), pp. 187–197. ISSN: 0018-9200. DOI: `10.1109/4.551910`.

[30] A. El Gamal and H. Eltoukhy. „CMOS image sensors“. In: *Circuits and Devices Magazine, IEEE* 21.3 (2005), pp. 6–20. ISSN: 8755-3996. DOI: `10.1109/MCD.2005.1438751`.

[31] E. Fossum. „CMOS image sensors: electronic camera-on-a-chip“. In: *Electron Devices, IEEE Transactions on* 44.10 (1997), pp. 1689–1698. ISSN: 0018-9383. DOI: `10.1109/16.628824`.

[32] G. Healey and R. Kondepudy. „Radiometric CCD camera calibration and noise estimation“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* (1994), pp. 267–276. ISSN: 0162-8828.

[33] Y. Tsin, V. Ramesh, and T. Kanade. „Statistical calibration of CCD imaging process“. In: *Proceedings of the Eighth International Conference on Computer Vision, (ICCV)*. Vol. 1. 2001, pp. 480–487.

[34] Y. Reibel, M. Jung, et al. „CCD or CMOS camera noise characterization“. In: *Eur. Phys. J* 21.21 (2003), pp. 75–80.

[35] M. Granados, B. Ajdin, et al. „Optimal HDR reconstruction with linear digital cameras“. In: *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE. 2010, pp. 215–222.

[36] A. Mansouri, F. Marzani, and P. Gouton. „Development of a protocol for CCD calibration: application to a multispectral imaging system“. In: *International Journal of Robotics and Automation* 20.2 (2005), pp. 94–100.

[37] H. Burger, B. Schölkopf, and S. Harmeling. „Removing noise from astronomical images using a pixel-specific noise model“. In: *International Conference on Computational Photography (ICCP).* IEEE. 2011, pp. 1–8.

[38] M. Goesele, W. Heidrich, and H. Seidel. „Entropy-based dark frame subtraction“. In: *Image Processing, Image Quality, Image Capture, Systems Conference 2001 (PICS), Proceedings of.* 2001, pp. 293–298.

[39] M. Gomez-Rodriguez, J. Kober, and B. Schölkopf. „Denoising photographs using dark frames optimized by quadratic programming“. In: *Computational Photography (ICCP), 2009 IEEE International Conference on.* 2009, pp. 1 –9. DOI: 10.1109/ICCPHOT.2009.5559013.

[40] A. Torralba and A. Olivia. „Statistics of natural image categories“. In: *Network: Computation in Neural Systems* 14.3 (2003), pp. 391–412. DOI: 10.1088/0954-898X_14_3_302. eprint: http://informahealthcare.com/doi/pdf/10.1088/0954-898X_14_3_302. URL: http://informahealthcare.com/doi/abs/10.1088/0954-898X_14_3_302.

[41] B. A. Olshausen and D. J. Field. „Sparse coding with an overcomplete basis set: A strategy employed by V1?“ In: *Vision research* 37.23 (1997), pp. 3311–3325.

[42] A. Hyvärinen, P. Hoyer, and M. Inki. „Topographic ICA as a Model of Natural Image Statistics“. English. In: *Biologically Motivated Computer Vision.* Ed. by S.-W. Lee, H. H. Bülthoff, and T. Poggio. Vol. 1811. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2000, pp. 535–544. ISBN: 978-3-540-67560-0. DOI: 10.1007/3-540-45482-9_54. URL: http://dx.doi.org/10.1007/3-540-45482-9_54.

[43] P. Arbelaez, M. Maire, et al. „Contour Detection and Hierarchical Image Segmentation“. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 33.5 (May 2011), pp. 898–916. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2010.161. URL: http://dx.doi.org/10.1109/TPAMI.2010.161.

[44] J. Deng, W. Dong, et al. „ImageNet: A Large-Scale Hierarchical Image Database“. In: *CVPR09*. 2009.

[45] The Astrophysical Research Consortium. *The Sloan Digital Sky Survey Project Book*. 1999. URL: http://www.astro.princeton.edu/PBOOK/ (visited on 09/17/2013).

[46] D. G. York, J. Adelman, et al. „The Sloan Digital Sky Survey: Technical Summary“. In: *The Astronomical Journal* 120.3 (2000), p. 1579. URL: http://stacks.iop.org/1538-3881/120/i=3/a=1579.

[47] J. E. Gunn, M. Carr, et al. „The Sloan Digital Sky Survey Photometric Camera“. In: *The Astronomical Journal* 116 (Dec. 1998), pp. 3040–3081. DOI: 10.1086/300645. eprint: arXiv:astro-ph/9809085.

[48] S. Binnewies and J. Pöpsel. *Capella Observatory Website*. http://www.capella-observatory.com/. Last accessed: 2013-10-05.

[49] D. J. Coffin. *Decoding raw digital photos in Linux*. http://www.cybercom.net/~dcoffin/dcraw/.

[50] M. Aharon, M. Elad, and A. Bruckstein. „K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation“. In: *IEEE Transactions on Signal Processing (TIP)* 54.11 (2006), pp. 4311–4322.

[51] A. Levin and B. Nadler. „Natural Image Denoising: Optimality and Inherent Bounds“. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2011.

[52] A. Levin, B. Nadler, et al. „Patch complexity, finite pixel correlations and optimal denoising“. In: *European Conference on Computer Vision (ECCV)*. 2012.

[53] H. C. Burger, C. J. Schuler, and S. Harmeling. „Image denoising with multi-layer perceptrons, part 2: training trade-offs and analysis of their mechanisms“. In: *arXiv:1211.1552* (2012).

[54] G. Cybenko. „Approximation by superpositions of a sigmoidal function“. In: *Mathematics of Control, Signals, and Systems (MCSS)* 2.4 (1989), pp. 303–314.

[55] M. Leshno, V. Lin, et al. „Multilayer feedforward networks with a nonpolynomial activation function can approximate any function“. In: *Neural networks* 6.6 (1993), pp. 861–867.

[56] K. Funahashi. „On the approximate realization of continuous mappings by neural networks“. In: *Neural networks* 2.3 (1989), pp. 183–192.

[57] A. Krizhevsky, I. Sutskever, and G. Hinton. „Imagenet classification with deep convolutional neural networks“. In: *Advances in Neural Information Processing Systems 25.* 2012, pp. 1106–1114.

[58] G. B. Orr and K.-R. Müller. *Neural Networks: Tricks of the Trade.* Springer-Verlag, 1998.

[59] Y. LeCun, L. Bottou, et al. „Efficient backprop“. In: *Neural networks: Tricks of the trade* (1998), pp. 546–546.

[60] J. Duchi, E. Hazan, and Y. Singer. „Adaptive Subgradient Methods for Online Learning and Stochastic Optimization“. In: *J. Mach. Learn. Res.* 12 (July 2011), pp. 2121–2159. ISSN: 1532-4435. URL: `http://dl.acm.org/citation.cfm?id=1953048.2021068`.

[61] M. D. Zeiler. „ADADELTA: An Adaptive Learning Rate Method“. In: *ArXiv e-prints* (Dec. 2012). arXiv:`1212.5701 [cs.LG]`.

[62] Moravian Instruments. *G2 CCD Camera Operating manual.* Version 2.4. Moravian Instruments. Czech Republic, 2010.

[63] D. Erhan, A. Courville, and Y. Bengio. *Understanding Representations Learned in Deep Architectures.* Tech. rep. 1355, Université de Montréal/DIRO, 2010.

[64] Y. Bengio and X. Glorot. „Understanding the difficulty of training deep feedforward neural networks“. In: *Proceedings of AISTATS.* Vol. 9. 2010, pp. 249–256.