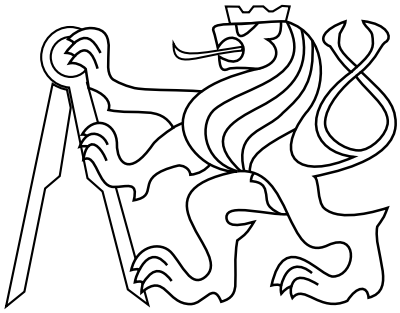
Czech Technical University in Prague

Faculty of Electrical Engineering
Department of Cybernetics
Multi-robot Systems



Intelligent Automatic Camera Exposure Control for Robot Sensing

Bachelor's Thesis

Tobiáš Vinklárek

Prague, May 2024

Study programme: Cybernetics and Robotics

Supervisor: Ing. Matouš Vrba

Acknowledgments

First and foremost, I would like to thank my supervisor, Ing. Matouš Vrba, for his support, helpful advice, and constructive feedback.

I also thank Ing. Viktor Walter for his troubleshooting assistance and the Multi-robot Systems Group group for providing the equipment necessary to conduct the experimental evaluation.

Last but not least, I would like to thank my family and friends for their continuous support throughout my studies.

BACHELOR‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Vinklárek Tobiáš** Personal ID number: **507641**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Cybernetics**

Study program: **Cybernetics and Robotics**

II. Bachelor’s thesis details

Bachelor’s thesis title in English:

Intelligent Automatic Camera Exposure Control for Robot Sensing

Bachelor’s thesis title in Czech:

Inteligentní řízení expozice kamery pro nasazení na robotech

Guidelines:

1. Research state-of-the-art intelligent methods of automatic exposure and gain control for cameras.
2. Select and implement several suitable solutions for deployment onboard unmanned aerial vehicles.
3. Experimentally validate the implemented solutions and compare them to the camera‘s integrated algorithm during real-world deployment of an aerial robot using suitable metrics.
4. Propose a specialized exposure and gain control method for improving detection of a single moving object within an image assuming a known bounding box. Implement, test, and compare this method to general exposure and gain control methods.

Bibliography / sources:

[1] N. Sampat, S. Venkataraman, T. Yeh, R. Kremens, "System implications of implementing auto-exposure on consumer digital cameras," Sensors, Cameras, and Applications for Digital Photography, pp. 100-107, 1999.
[2] U. Shin, J. Park, G. Shim, F. Rameau, I. S. Kweon, "Camera Exposure Control for Robust Robot Vision with Noise-Aware Image Quality Assessment," IROS, pp. 1165-1172, 2019.
[3] E. Onzon, F. Mannan, F. Heide, "Neural Auto-Exposure for High-Dynamic Range Object Detection," CVPR, pp. 7710-7720, 2021.
[4] I. Shim, T. -H. Oh, J. -Y. Lee, J. Choi, D. -G. Choi and I. S. Kweon, "Gradient-Based Camera Exposure Control for Outdoor Mobile Platforms," Transactions on Circuits and Systems for Video Technology, vol. 29, no. 6, pp. 1569-1583, 2019.

Name and workplace of bachelor’s thesis supervisor:

Ing. Matouš Vrba Multi-robot Systems FEE

Name and workplace of second bachelor’s thesis supervisor or consultant:

Date of bachelor’s thesis assignment: **12.01.2024** Deadline for bachelor thesis submission: **24.05.2024**

Assignment valid until: **21.09.2025**

Ing. Matouš Vrba	prof. Dr. Ing. Jan Kybic	prof. Mgr. Petr Páta, Ph.D.
Supervisor’s signature	Head of department’s signature	Dean’s signature

III. Assignment receipt

The student acknowledges that the bachelor’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor’s thesis, the author must state the names of consultants and include a list of references.

.	
Date of assignment receipt	Student’s signature

Declaration

I declare that the presented work was developed independently, and that I have listed all sources of information used within, in accordance with the Methodical instructions for observing ethical principles in preparation of university theses.

Date ....

Abstract

In autonomous robotics, the properties of an onboard camera are crucial for the successful execution of numerous robotic tasks. The most important property is the camera's image quality, which directly affects performance of the tasks relying on it. One of the primary factors influencing said quality is the selection of an appropriate exposure time of the camera. In dynamic conditions, to ensure optimal image quality through accurate exposure, cameras utilize Automatic Gain Control (AGC) and Automatic Exposure Control (AEC) to determine the ideal exposure time. Camera manufacturers, however, frequently use simplistic AGC and AEC techniques, which can lead to a reduction in image quality and the incorrect functioning or failure of the running onboard task, by not adjusting the cameras' exposure and gain correctly. This thesis discusses the topic of camera control, explores current State-Of-The-Art (SOTA) AEC strategies, and suggests a simple AEC method with feedback from an onboard object detector. Several strategies are implemented and their exposure control ability is evaluated. Based on these preliminary results, several algorithms are then selected and deployed on board a Micro Aerial Vehicle (MAV) performing a detection task.

Keywords Automatic control, Computer vision, Unmanned aerial vehicles

Abstrakt

V autonomní robotice patří zpracování obrazu k nejdůležitějším disciplínám a je proto důležité zajistit vstupní data v dostatečné kvalitě k jejich správnému vykonání. Kvalita obrazu jako taková je primárně ovlivněna správným výběrem zesílení a doby expozice, které jsou předány kameře. Jestliže se ale prostředí dynamicky mění, je nutné zesílení a dobu expozice kontinuálně upravovat. Tyto úpravy jsou zajišťovány algoritmy Automatického Řízení Expozice (AŘE) a Automatického Řízení Zesílení (AŘZ). Výrobci kamer však často používají algoritmy AŘE a AŘZ, které nejsou dostatečně robustní, což může vést k přeexponování nebo podexponování obrazu a selhání na něm závislých disciplín, jako například detekce či lokalizace. Tato práce se zabývá základním rozborem řízení kamery, následně představuje a implementuje některé aktuální nejpokročilejší metody AŘE spolu s vlastní metodou řízení AŘE, která bere v potaz zpětnou vazbu z detekční úlohy. Implementované algoritmy jsou prvně experimentálně otestovány pro jejich schopnost řízení expozice a pomocí metrik je vyhodnocena kvalita obrazu řízených kamer a porovnána s vestavěným AŘE kamery. Dle výsledků jsou pak vybrané algoritmy nasazeny na bezpilotní letecký prostředek vykonávající detekční úlohu. Data z tohoto experimentu jsou posléze empiricky vyhodnocena a je ukázáno, že přidání zpětné vazby z detekční úlohy do řízení expozice kamery vede ke zlepšení detekce.

Klíčová slova Automatické řízení, Počítačové vidění, Bezpilotní letouny

Abbreviations

MRS Multi-robot Systems Group

UGVs Unmanned Ground Vehicles

MAV Micro Aerial Vehicle

MAVs Micro Aerial Vehicles

VO Visual Odometry

SLAM Simultaneous Localization And Mapping

SOTA State-Of-The-Art

CNN Convolutional Neural Network

AEC Automatic Exposure Control

AGC Automatic Gain Control

MOS Metal Oxide Semiconductor

CMOS Complementary Metal Oxide Semiconductor

CCD Charge-Coupled Device

SWaP Size, Weight and Power

Contents

1 Introduction

Since humans have been able to transfer and display captured images in digital form, camera digital image processing has been an important area of computer science. It has received increased attention in recent years, primarily because of advancements in autonomous driving and artificial intelligence that use camera imaging technology for tasks like image classification, semantic segmentation, image captioning, pose estimation, image enhancement, and more. If these tasks receive images of inadequate quality, it can lead to failure in their execution and potential loss of equipment or life.

Many manufacturers of digital cameras use simple, reliable AEC and AGC algorithms, which are sufficient for the majority of autonomous tasks. Nevertheless, these cameras frequently reach their hardware limits when used in high dynamic range environments (settings with significant brightness variation) and are unable to deliver sufficient image quality for the tasks listed to be completed correctly.

The resulting images frequently have either an excessively high or low exposure (see Figure 1.1), which severely restricts or eliminates the ability to extract the necessary information from them. However, an image that appears excessively exposed or inadequately exposed to humans may be correctly exposed for computer vision or image processing task.

(a) Example of an overexposed image.

(b) Example of an underexposed image.

Figure 1.1: Examples of scenes with bad exposure from the perspective of a human eye.

In this thesis, several SOTA AEC and AGC methods were researched, implemented, and compared along with a rapid and lightweight control algorithm that takes feedback from onboard detection into account. The verification of implemented methods was conducted in two separate experiments. The first experiment served as preliminary testing of all implemented methods together with built-in AEC and AGC to see how well they perform in an outdoor environment, and whether their control of exposure time had desired properties, such as response time to incoming change or stability. Based on the results of the first experiment, the second experiment was conducted onboard a MAV running an object detector to determine if the implementations provide an improvement in detection compared to built-in AEC and AGC. Both experiments are evaluated, and the results are presented.

1.1 Related works

Even though camera exposure control is a long-standing area of study, advances in technology in the past two decades made it feasible to implement this problem in ways that require more processing power, and are not integrated into the camera itself. [1] is frequently cited work providing an understanding of the AEC and AGC methodologies being used by the industry, most of them involving light metering sensors and weighted averaging of images. It is important to note, that this work was released nearly 30 years ago, and current approaches can be different. For example, the cameras used in this work, introduced in subsection 4.1.1, use a binary search algorithm to locate the ideal exposure time.

Nearly all methods state an improvement of an onboard task relying on the camera images as the primary reason for their motivation. Developing an algorithm that enhances a Visual Odometry (VO) or Simultaneous Localization And Mapping (SLAM) is one such popular field [2]–[7].

The simplest approach, besides setting a fixed exposure time, is to use a P regulator to control the exposure time. For this, the error value is determined by averaging intensity of pixels in an image following the application of masks [8]–[10]. This simplicity is a compromise for evaluation speed.

Some techniques employ one or more look-up tables to anticipate brightness change given a new exposure in an effort to maximize speed [11], [12]. However, these works only show a minor sample of useful test findings.

More sophisticated methods try to assess the input images based on factors relevant to the onboard task, like entropy [5], [13], gradient [2], [14]–[16], their combination [3], [17], or intensity, mapped as a function [7], [18]. The subsequent exposure time and gain are obtained by an optimization problem provided with the image evaluation. These approaches frequently deliver results that are well-balanced between complexity and effectiveness.

Deep learning and neural network-based techniques have become more prevalent in the recent years [6], [19]–[21]. Their primary drawback is that they are importable, meaning that altering how onboard tasks are carried out would necessitate total network retraining. While they can produce satisfactory outcomes with fast calculation times, they typically require additional hardware, such as GPU, which can be inconvenient when employed on a MAV.

Very few studies [19], [21] include input from onboard computer vision tasks, such as detection or feature matching, into their exposure control. To the best of your knowledge, none of such studies provides a method that would not assume a very specific task.

Several of the approaches listed in this section are explored and then implemented, compared, and assessed in this work. In addition to that, a relatively universal feedback loop from the onboard detection task into a AEC and AGC control system is designed, showing that a simple method can yield better results when given additional data.

2 Principles of camera operation

This chapter provides a brief overview of the image acquisition and processing procedure of a typical video camera.

2.1 Mechanical aspects

Image sensor

The photosensitive image sensor, which converts incoming light into a usable electrical signal before being further processed to create an image, is a crucial component of a camera. Charge-Coupled Device (CCD) and active-pixel sensors (also known as Complementary Metal Oxide Semiconductor (CMOS) sensors) are the two primary types of sensors. Based on Metal Oxide Semiconductor (MOS) technology, each sensor offers a distinctive set of advantages for its targeted application, such as production cost, noise, speed, dynamic range or power consumption [22], [23].

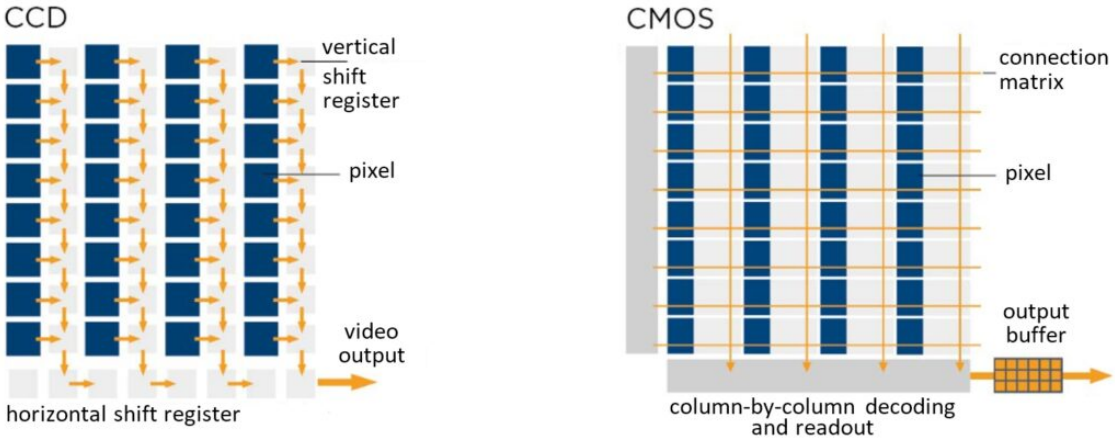


Figure 2.1: Architecture comparison between CMOS and CDD sensor. In the CCD sensor, the charge is shifted pixel by pixel and read out at the end. In the CMOS sensor is read directly, increasing speed. Image taken from Astrobasics website[1].

The size of the image sensor is another crucial factor. Together with pixel size, it determines the image resolution and affects the depth of field, noise, low-light performance, and other factors. It is also crucial to consider what color space the sensor covers and what color depth (bit depth) it can achieve [23].

Optical aperture

The optical aperture is another factor that influences the quality of an image. It describes the opening through which light passes on its way to the camera's image sensor.

The size of an aperture affects the depth of field and exposure time. Larger apertures achieve lower exposure times at the cost of shallower depth of field, while smaller apertures

[1]Source: https://astrobasics.de/en/components/camera/ccd-or-cmos/

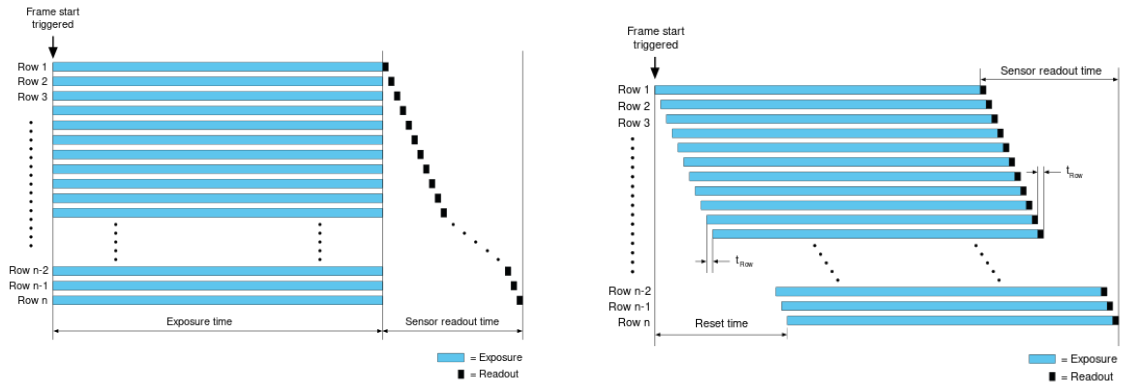
achieve high depth of field, but require longer exposure time, which is necessary to achieve the same image brightness.

The aperture typically includes a lens, which alters the image's distortion and field of view. A polarization filter can also be added to the lens to increase contrast, and color saturation, or reduce the effect of reflections, though this will result in less light passing through the aperture, thus requiring a longer exposure time.

Camera shutter

A camera shutter [24] is a part of camera that controls the duration for which the image sensor is exposed to light. The camera's shutter may be electrical, mechanical, or both. The majority of modern digital video cameras rely on an electronic shutter that is managed by the camera control unit rather than a mechanical shutter. This gives the start and stop triggers a negligible delay. Electronic shutters are primarily of two types:

a) Global shutter
   All sensor row values are reset upon the acquisition trigger, initiating a fresh exposure cycle. All sensor rows halt the exposure after the exposure time has passed, and the camera then reads out each row one after the other. Global shutters require a more complex design than rolling shutters, which increases their cost, however, they benefit from precise image synchronization which is essential for capturing fast motion.
b) Rolling shutter
   The camera resets and begins exposing the first row upon receiving the acquisition trigger. With an offset approximately comparable to the read-out time, the subsequent row exposure is always initiated after the preceding one. The same exposure time is given to each row, and the results are read out immediately after finishing. Rolling shutter is not suitable for capturing fast-moving objects as the sequential exposure and readout distort the image. However, besides lower cost and power-effectiveness, it allows the camera to capture images at higher frame rate.



(a) Global shutter (b) Rolling shutter

Figure 2.2: Examples of shutter types. Image taken from Basler website[1].

[1]Source: https://docs.baslerweb.com/electronic-shutter-types

2.2 Digital operations

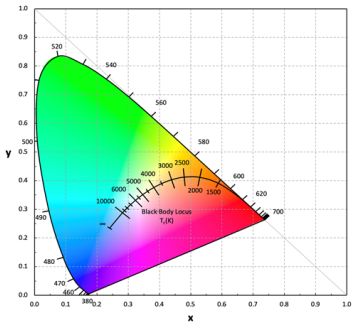
Many procedures are frequently used to improve an image's final quality either while or after it is captured in an effort to make up for undesirable effects of the environment in which the scene is shot as well as inherent flaws in the camera hardware.

Exposure time

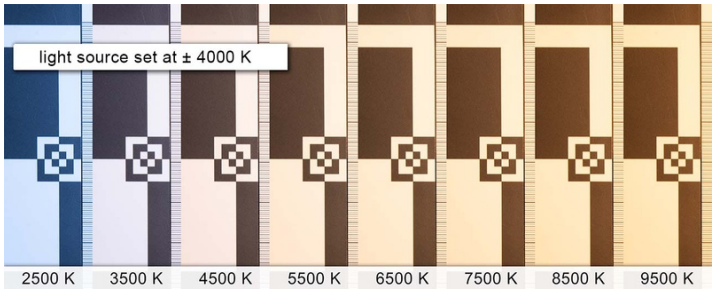
Exposure time is the duration of the image sensor capturing the incoming light. It is one of the crucial values in camera control. If it is set incorrectly, the image's information ca be irreversibly lost. Additionally, it sets a restriction on the camera's maximum frame rate, which cannot be higher than the inverse of exposure time.

Gain

In order to maintain desirable image characteristics like brightness or frame rate, a camera with insufficient sensor sensitivity may artificially boost the brightness of the image when it captures a dimmer scene. It can be accomplished by amplifying the signal that is coming in from the camera sensor by a certain factor called gain. However, gain also amplifies flaws like noise, making it more prominent. Consequently, the used gain is typically desired to be minimal.



(a) The black body locus on a chromaticity diagram. Image taken from Luminus[1].



(b) Example of how different camera color temperature settings affect the outcoming image. Image taken from Fstoppers[2].

Figure 2.3: Color temperature in color space (left) and how it affects the characteristics of an image (right).

White balance

Light from various sources contains different wavelength mixtures, which are referred to as color temperatures. Color temperature is defined by a curve on a chromaticity diagram that represents the color of light emitted by an ideal black body at various temperatures, as can be seen in Figure 2.3a. The human eye has an ability to adapt to these changes, but cameras may have trouble accurately interpreting the light in the scene, resulting in an image color distribution that deviates from the actual situation, as can be seen in Figure 2.3b. Automatic

[1]Source: https://fstoppers.com/natural-light/there-difference-between-color-temperature-and-white-balance-596031

[2]Source: https://luminusdevices.zendesk.com/hc/en-us/articles/4403685063437-What-do-CCT-CIE-and-SPD-mean-in-LED-lighting

white balance algorithms [25], [26] are commonly used in modern video cameras, allowing for real-time adjustments and counteracting the adverse impacts of light sources. However, problems can arise, when several light sources with different color temperatures are present in the scene.

Color balance

As with white balance, which is occasionally seen as a subset of color balance, the main focus is on the adjustment of the image's red, green, and blue channels [27], [28]. However, rather than moving along a curve on a chromaticity diagram like white balance, color balance can adjust each color channel independently, similarly as in Gain. The adjustment is usually done manually by the user. In computer vision, color balance or white balance is not universally required or desired, as it depends on the specific requirements of the task. An incorrect choice can potentially lead to a loss of information contained in the image, for example through grayscale conversion [29].

Dark current

Due to reverse-bias leakage in diodes, a tiny current flows through the sensor even when no photons are reaching the photosensitive image sensor, producing fixed-pattern noise in the image. This is typically eliminated through digital post-processing, such as dark-frame subtraction, or it can also be reduced by a creative sensor construction [30], [31].

Noise reduction

Though steps can be taken to mitigate it, noise comes from a variety of sources, including the previously mentioned dark current and gain. As such, it is something that can never be eliminated. Thus, cameras use a wide range of various noise reduction techniques and algorithms in image post-processing to assure at least a reduction, even at the expense of small image distortion. There are numerous approaches to this topic, such as mathematical morphology [32], median filtering [33], fuzzy filtering [34], weighted averaging [35], [36] or spatiotemporal smoothing [37].

3 Control algorithms

This chapter introduces several SOTA AEC methods, some of which were implemented and evaluated. The methods that were implemented are described in more detail.

3.1 Image gradient-based methods

These exposure control algorithms exploit gradient information contained in an image, to highlight extractable features while suppressing uniform areas that are redundant. Additionally, they introduce metrics to evaluate the image gradient quality and achieve better exposure control.

Gamma correction mapping

Shim et al. employ one such strategy in [14], where they introduce an evaluation metric along with functions that calculate the desired exposure time. Subsequently, they expand on these foundations in [4] by adding exposure control to a multi-camera system and introducing a non-linear function to calculate the new exposure time.

The image's normalized gradient magnitude serves as the foundation for the evaluation metric itself. A threshold is included to disregard minor gradients in the final evaluation because the majority of the gradients in the image have small values, and they desire to focus on strongly defined gradients. The evaluation metric M is defined as

$$\bar{m}_{\mathrm{i}} = \begin{cases} \dfrac{\log(\lambda(m_{\mathrm{i}} - \delta) + 1)}{\log(\lambda(1 - \delta) + 1)} & \text{if } m_{\mathrm{i}} \geq \delta, \\ 0, & \text{otherwise,} \end{cases} \tag{3.1}$$

$$M = \sum_{\mathrm{i}} \bar{m}_{\mathrm{i}}, \tag{3.2}$$

where m_{i} is the normalized gradient magnitude of the image in range $\langle 0; 1 \rangle$, $\bar{m}_{\mathrm{i}}$ is the amount of gradient information corresponding to gradient magnitude, δ is the activation threshold, and λ is a control parameter. Both λ and δ are parameters adjustable by the user.

This provides an evaluation of quality of the current image, which is to be maximized. To get information about direction and magnitude of the new exposure update, the method introduces an algorithm based on γ-correction, which for $\gamma < 1$ increases and for $\gamma > 1$ decreases the image brightness. The definition of γ-correction is

$$I_{\mathrm{in}}^{\gamma} = \left(\frac{I_{\mathrm{in}}}{255} \right)^{\gamma} \cdot 255, \tag{3.3}$$

where I_{in} is the original image (incoming image), γ is the gamma correction power to which the image is raised and I_{in}^{γ} is the image with applied gamma correction.

This is used in the γ-correction mapping evaluation, described as

$$\hat{\gamma} = \arg\max_{\gamma}(M(I_{\mathrm{in}}^{\gamma})), \tag{3.4}$$

where $\hat{\gamma}$ is the optimal value of γ.

In order to solve Equation 3.3, several anchor points with predetermined γ values are created, which are then fitted by a fifth-degree polynomial. The optimal $\hat{\gamma}$ is then obtained by finding the maximum of the fitted polynomial on an interval $[0.5; 2]$, empirically determined by Shim. This should, in theory, ensure the methods' smoothness while maintaining acceptable computational time.

The original paper [14] proposes a linear update function to control the camera exposure time, defined as

$$\alpha = \begin{cases} 1/2, & \hat{\gamma} \geq 1, \\ 1, & \text{otherwise}, \end{cases} \tag{3.5}$$

$$E_{\mathrm{t+1}} = (1 + \alpha K_{\mathrm{p}}(1 - \hat{\gamma}))E_{\mathrm{t}}, \tag{3.6}$$

where α is a control parameter used to decrease the steepness of descent, E_{t} is the current exposure time, $E_{\mathrm{t+1}}$ is the new exposure time and K_{p} is an adjustable gain parameter to control the speed of convergence.

The author's subsequent work presents a non-linear update function and shows that it has better convergence than the linear update function [4]. The definition of the non-linear update function is

$$E_{\mathrm{t+1}} = (1 + \alpha K_{\mathrm{p}}(R - 1))E_{\mathrm{t}}, \tag{3.7}$$

$$R = d \cdot \tan\left\{(2 - \hat{\gamma}) \cdot \arctan\left(\frac{1}{d}\right) - \arctan\left(\frac{1}{d}\right)\right\} + 1, \tag{3.8}$$

where d is an adjustable parameter.

This method's primary drawback is that it relies on gradient information, which is very low in homogeneous images (completely black or white). The method fails to control the exposure if such a state is reached, and the authors do not mention this scenario or elaborate on how to avoid it.

Gradient ascent

Zhang et al. provide an alternative gradient-based method [15], which is often compared with Shim [14]. Similar to [14], Zhang et al. present a unique metric for assessing the quality of an image and demonstrates that the metric performs better than the one presented in [14]. However, rather than using it to evaluate the image, the approach from [15], uses this metric to establish an optimization problem using gradient ascent, which then yields the new desired exposure time.

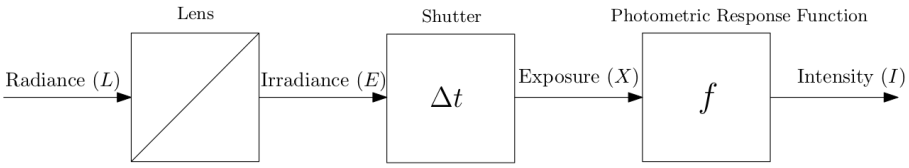


Figure 3.1: A visualization of the effect of the scene radiance on the resulting image brightness. Image taken from [15].

At first, a photometric response function is established [38], which is the outcome of the image creation process (illustrated in Figure 3.1). The photometric function is defined as

$$X = E\Delta t, \tag{3.9}$$
$$f(X) = f(E\Delta t) = I, \tag{3.10}$$

where E is the scene irradiance, which is the amount of energy that reaches the camera sensor per time unit, Δt is the current exposure time, which, when combined with E, gives the total amount of energy X that the sensor receives during an exposure cycle. Upon completing the internal processing outlined by function f, the image brightness I is obtained.

For convenience, since the photometric function f is invertible and grows monotonically, the natural logarithm of inverse response function g can be defined as

$$f^{-1}(I) = E\Delta t, \tag{3.11}$$
$$\ln\{f^{-1}(I)\} = \ln(E) + \ln(\Delta t), \tag{3.12}$$
$$g(I) = \ln\{f^{-1}(I)\}. \tag{3.13}$$

The exposure control problem is then formulated as maximization of a gradient-based metric $M_{softperc}$, defined as

$$M_{softperc}(p) = \sum_{i=0}^{S} W(i) \cdot G(i), \tag{3.14}$$
$$W(i) = \begin{cases} \frac{1}{N}\sin\left(\frac{\pi \cdot i}{2(p \cdot S)}\right)^k, & i \leq p \cdot S, \\ \frac{1}{N}\sin\left(\frac{\pi}{2} - \frac{\pi(i-(p \cdot S))}{2(S-(p \cdot S))}\right)^k, & \text{otherwise}, \end{cases} \tag{3.15}$$
$$N = \sum_{i=0}^{S} W(i), \tag{3.16}$$

where $G(i)$ represents gradient magnitudes, obtained via the Sobel operator, sorted in ascending order, and i represents the i-th index of G and k the scaling factor. The number of pixels in the image is denoted S. The weight function $W(i)$ emphasizes values of the sorted gradient magnitudes and is dependent on the set percentile p. N serves as a normalization factor of the weight function. The effect of parameter p on the outcome of the weight function can be seen in Figure 3.2, where $S = 99$ and $k = 7$.

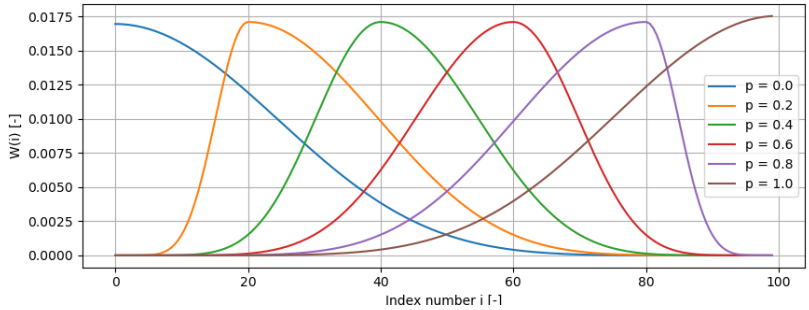


Figure 3.2: Behavior of a weight function $W(i)$ based on p.

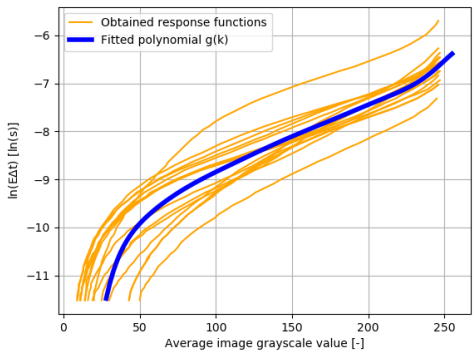
The gradient ascent optimization method, which determines the next optimal exposure time Δt_{next}, can be used to maximize the metric (3.14). The complete method for obtaining the formula is explained in [15], and results in

$$\frac{\partial G(\cdot)}{\partial \Delta t} = 2[\nabla I(\cdot)]\nabla \left[\frac{1}{g'(I(\cdot))\Delta t}\right]^T, \tag{3.17}$$

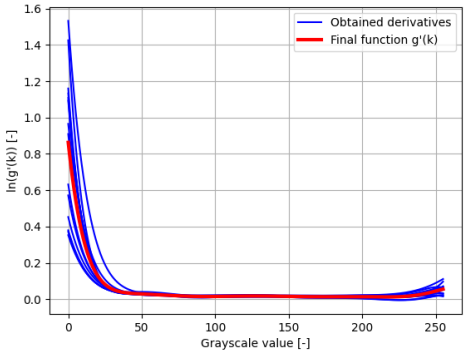
$$\frac{\partial M_{\text{softperc}}}{\partial \Delta t} = \sum_{i=0}^{S} W(i)\frac{\partial G(i)}{\partial \Delta t}, \tag{3.18}$$

$$\Delta t_{\text{next}} = \Delta t + \gamma \frac{\partial M_{\text{softperc}}}{\partial \Delta t}, \tag{3.19}$$

where γ controls the step size, Δt represents the current exposure time in seconds and $(\cdot)$ means 'apply to all pixels'. Equation (3.17) utilizes the image gradient and the derivative of the inverse photometric response function g.



(a) An example of fitting polynomial function on a dataset obtained on a sunny day.



(b) A plot of the derivates of all fitted polynomials from all datasets together with the final approximation function.

Figure 3.3: An example of recovering g for one dataset and obtaining the final approximation of the function g'.

The requirement for obtaining a valid inverse photometric response function g is a scene with constant illumination. To estimate the function, the camera exposure time is set at its lowest possible value, and is then raised exponentially (linearly in case the camera response

is linear and not logarithmic), while recording the image's grayscale value for each unique exposure time, yielding a set of points representing the relationship between exposure time and image brightness, which is then fitted by a tenth-degree polynomial to obtain a continuous function representing the estimation of g.

To obtain an accurate representation of g in a specific environment, such as a sunny day, the measurement is repeated multiple times at different locations to obtain coefficients of multiple functions g, completing a single dataset. Averaging the coefficients of a given dataset presents a reasonable estimation of the function g for the given environment. Figure 3.3a shows an example of estimating function g from a dataset.

(a) 24 μs (b) 103 μs (c) 215 μs (d) 455 μs

Figure 3.4: Example scenes from one of the datasets obtained on a sunny day at different exposure times.

In total, over 200 different scenes split into 19 datasets were captured with two cameras, producing 19 diverse approximations of the function g. The final approximation of function g' was then obtained by least-squares fitting the derivatives of the 19 approximations of function g. Figure 3.3b shows the retrieval of the final approximation of function g'.

This can be then applied in equation (3.19) to determine the next exposure time, where for each pixel value in I, the derivative of the photometric response function is calculating using the formula $g'(I(\cdot))$. As a result, equation (3.17) returns a two element vector for each pixel that describes the gradient in the x and y axes. A scalar product is then used to get the derivative of a particular pixel. In order to obtain $\frac{\partial G(i)}{\partial \Delta t}$, the pixel derivatives are ordered in ascending order and together with $W(i)$, $\frac{\partial M_{\text{softperc}}}{\partial \Delta t}$ can be computed.

Theoretically, both gradient-based methods should settle at approximately the same exposure times when operating in parallel. This means that the primary areas where the approaches should differ are response time and robustness. [15], in contrast to [14], deals with the issue of an entirely overexposed or underexposed picture, by adding -2 for overexposed pixels and 2 for underexposed pixels to the gradient when a pixel's gradient equals zero, which indicates that the pixel and its surroundings are fully saturated. This should guarantee that the algorithm handles saturated images correctly.

3.2 Extended gradient methods

The methods described in this section employ not only the use of image gradients to determine image quality but also additional metrics to introduce additional information into the evaluation.

Gradient, noise and entropy

Shin et al. [17] present a method that builds on the gradient method introduced by [14] (discussed in section 3.1.1) by adding additional evaluation metrics to compute the overall image quality. This information can then be used to determine the next exposure time using an optimization method. The suggested gradient-based metric is

$$G = \{G_1; G_2; \ldots; G_{N_\mathrm{C}}\}, \tag{3.20}$$

$$G_\mathrm{j} = \sum_{\mathrm{i} \in C_\mathrm{j}} \bar{m}_\mathrm{i}, \; j = 1, 2, .., N_\mathrm{C}, \tag{3.21}$$

$$L_\mathrm{gradient} = K_\mathrm{g} \cdot E(G)/s(G), \tag{3.22}$$

where $\bar{m}_\mathrm{i}$ denotes the gradient calculated using equation (3.1). [17] divide the obtained gradient image into a total of N_C segments. C_j represents the number of pixels in each segment j and is used to calculate the segment's value. By computing the mean and standard deviation of G, represented here by operators $E(\cdot)$ and $s(\cdot)$, respectively, the final gradient metric evaluation is obtained.

In addition to the enhanced gradient-based metric, the image's entropy and noise are assessed. The noise assessment metric is based on a filter methodology, and is defined as

$$L_\mathrm{noise} = \sqrt{\frac{\pi}{2}} \frac{1}{N_\mathrm{S}} \sum_\mathrm{i} H(i) \circ U(i) \mid I * M \mid, \tag{3.23}$$

where N_s is the number of non-zero elements in $H \circ U$, $*$ is the convolution operator, $|\cdot|$ is the absolute value operator, and $\circ$ is the element-wise product, $H(i)$ and $G(i)$ are masks defined as

$$H(i) = \begin{cases} 1, & m_\mathrm{i} \leq \delta, \\ 0, & \text{otherwise}, \end{cases} \tag{3.24}$$

$$U(i) = \begin{cases} 1, & \tau_\mathrm{l} \leq I(i) \leq \tau_\mathrm{h}, \\ 0, & \text{otherwise}, \end{cases} \tag{3.25}$$

where δ is the p-th percentile of gradients in the image and m_i represents the values of normalized gradients. Thus, homogeneous sections in the image can be obtained by applying the mask H. U represents a mask of the unsaturated regions in the image, determined by the lower bound τ_l and upper bound τ_h. Since [17] assume that the noise in the picture is zero-mean additive Gaussian noise, M is used as the noise estimate kernel, and defined as

$$M = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}. \tag{3.26}$$

In case the image has multiple color channels, the noise level is estimated in each channel using this metric, and the results are averaged to obtain the final value.

The final metric to be introduced is designed to evaluate the image's low-level information based on entropy. It is defined as

$$L_\mathrm{entropy} = -K_\mathrm{e} \cdot \sum_{k=0}^{255} P_\mathrm{I}(k) \log_2(P_\mathrm{I}(k)), \tag{3.27}$$

where K_e is a normalization factor and $P_l(k)$ represents the likelihood that a given pixel value k appears in the image, defined as

$$P_l(k) = \frac{N_k}{N_x}, \tag{3.28}$$

where N_k is the number of pixels with value k in the image, and N_x is the total number of pixels in the image.

These three metrics are then combined with the user-defined weights α and β to generate the final image quality metric

$$f(I) = \alpha L_{\text{gradient}} + (1 - \alpha) L_{\text{entropy}} + \beta L_{\text{noise}}. \tag{3.29}$$

An image with a higher value of $f(I)$ contains lower noise levels and more strongly defined texture details. As a result, it is desired to maximize its value.

[17] opt to achieve that by utilizing the Nelder-Mead method [39], [40], which makes use of the concept of simplex and does not necessitate the use of a derivative. In Appendix B, the complete algorithm is explained. The simplex yields the optimal exposure time and gain for a particular scene with impact to $f(I)$ after convergence or an approximation after a given number of iterations.

This method's primary drawback is that it was designed for static scenes. To resolve this problem, our implementation periodically resets the simplex to provide it with new values.

3.3 Non-gradient methods

Unlike earlier approaches, these methods do not take advantage of the image gradient. To properly adjust exposure time, a feedback loop from the running onboard task utilizing the resulting images is introduced.

Neural networks

One of the latest State-Of-The-Art approaches [21] proposes a 1D Convolutional Neural Network (CNN) evaluation of raw image histograms, together with processing of image features extracted using the ResNet architecture, as shown on Figure 3.5.

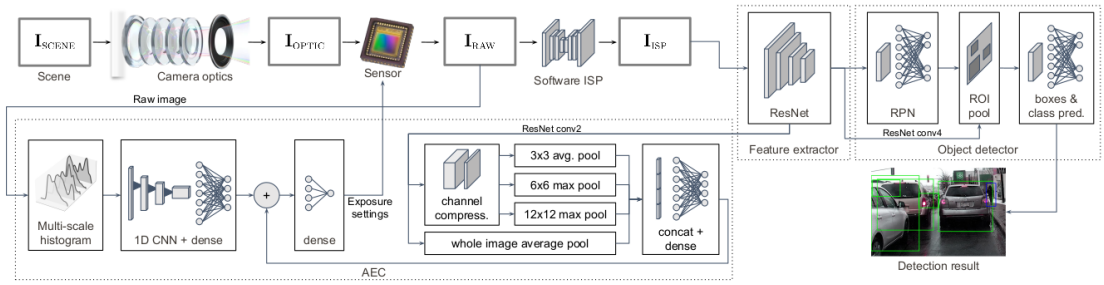


Figure 3.5: Over-view of the proposed CNN-based exposure control architecture with live object detection. Image taken from [21].

This method was not implemented due to its complexity and non-portability (it would have to be implemented on a unique camera hardware, limiting its usability). The technique is intended for usage on Unmanned Ground Vehicles (UGVs) equipped with an onboard GPU to attain acceptable computational times. However, this is severely limiting for aerial robots,

especially for Size, Weight and Power (SWaP) MAVs. Additionally, altering the detection's target object would necessitate starting over with a fresh dataset acquisition and network training, even further reducing its possible applications.

- **Detection area focus**

This method was designed as a part of this thesis. It uses a proportional regulator control algorithm and a feedback loop to improve the image quality after onboard detection. This ensures accurate target detection despite changes in surroundings conditions, until detection is no longer possible due to moving out of frame, moving too far away from the target, or due to too complex environmental conditions, (for example very quick changes in scene illumination).

If no detection is observed, the algorithm uses the P regulator to compute the average grayscale value of the picture and adjusts the exposure time based on the difference from the desired control value using the proportional regulator.

In the case of a detection, the method receives data from the detector in the form of at least four corner points defining a rectangle that encompasses the detected object. This region then defines the region of interest, for which a mask is generated. The mask is then enlarged and blurred using Gaussian Blur to include extra information about the target's immediate surroundings and compensate for a potentially inaccurate detection.

The regulator utilizes the average of the pixel values weighted by the mask defined as

$$i = \{0; 1; \ldots; N_\mathrm{I} - 1\}, \tag{3.30}$$

$$N_\mathrm{D} = \sum_i D(i), \tag{3.31}$$

$$E = \frac{1}{N_\mathrm{D}} \sum_i D(i) \cdot I(i), \tag{3.32}$$

where N_I is the number of pixels in the image, $D(i)$ is the generated mask, N_D is the sum of all weights in the mask $D(i)$, $I(i)$ is the same incoming image as used by the detector and E is the average of the pixel values weighted by the mask. From E and the control value, a deviation is calculated, which is then used by the proportional regulator to control the exposure time. In the case of detection, the deviation is quintupled to improve the response time to sudden illumination changes of the target.

This approach is designed to work with a single target that, once detected, remains detectable for as long as possible. It is easy to add the ability for multiple target detections. However, this raises the possibility of many targets with varying levels of illumination, which can lead to unexpected results and would require more complex implementation. The detection of multiple targets is outside the scope of this work.

4 Experiments

The conducted experiments, used cameras, and their parameters are elaborated on in this chapter, together with the environments in which the experiments were conducted, the evaluation of the obtained data, and the presentation of the results.

4.1 Setup & configuration

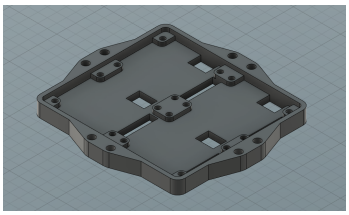
Used cameras & their parameters

For the experiments, two cameras were used. A Basler dart daA1600-60uc camera and Baluff Matrix Vision mvBlueFOX-MLC200wC. Both cameras were equipped with a fish-eye lens of similar properties.

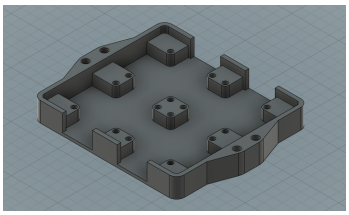
For both cameras, the region of interest was limited to 480×480 pixels, to mask out the MAV rotors from the image. Since the Basler camera offers a default resolution of 1600×1200 pixels, a binning of 2 was applied to reduce noise in the image, resulting in a resolution of 800×600 pixels. Every other image quality-enhancing option such as auto white balance was kept on. The camera frame rate limits were disabled. Thus, the frame rates were limited purely by the exposure time, camera readout, image processing, and bandwidth. The images were captured in the BGR8 pixel format. Parameters of the different AEC algorithms are listed in Appendix A and were constant for all experiments.

Camera holders

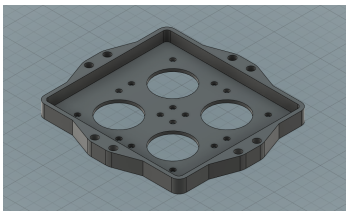
To ensure a fair comparison, the control algorithms run on separate cameras in parallel and the cameras are mounted as close to each other as possible, ensuring the perceived scene is nearly identical. To achieve that, the camera holders were modeled in a CAD software (see Figure 4.1) and 3D-printed. The holders were designed to be mounted onto the MAV.



(a) Bluefox camera holder.



(b) Basler camera holder - bottom part.



(c) Basler camera holder - top part.

Figure 4.1: Models of custom camera holders.

The design takes into consideration safety of the mounted cameras since both types come with exposed components that can be easily broken off under mechanical stress. Additionally, both holders are equipped with protective covers for the lenses to ensure the cameras are not damaged during transportation. The cameras mounted are shown at Figure 4.2.

(a) Bluefox camera holder with mounted cameras. (b) Basler camera holder with mounted cameras.

Figure 4.2: The camera holders with mounted cameras.

4.2 Preliminary testing

This section describes a set of tests that were conducted in order to determine performance and functionality of the implemented control algorithms and assess their usability onboard a MAV.

Configuration

The experiments, were conducted on a sunny day in a feature-rich environment of a park, in a location that offered a high brightness range. A photo of the testing area is presented in Figure 4.3. The evaluation datasets were captured on an approximately 50-meter-long circular path with the cameras facing forwards, which provided roughly a one minute of footage for each dataset. Since only three Bluefox cameras were available, the number of used Basler cameras was also reduced to three to provide an easy comparison.

Results

For the evaluation of each dataset, three feature extractors were used: FAST [41], Shi-Tomasi [42] and SIFT [43]. The used evaluation metric was the mean number of extracted features from each extractor is a given dataset. The captured images were also examined visually to assess the reaction to changes and stability of the control algorithms.

Label	Control method name
A	Built-in AEC & AGC control
B	Gamma correction mapping([4])
C	Gradient ascent ([15])
D	Gradient Noise Entropy ([17])
E	Detection area focus with no detection feedback

Table 4.1: The implemented and compared AEC and AGC methods.

	Basler FAST				
	A	B	C	D	E
park_1	**6578**	624	412	-	-
park_2	**6213**	3184	-	1840	-
park_3	**6287**	1135	-	-	2487
park_4	**5873**	-	608	3145	-
park_5	**5719**	-	1574	-	2699
park_6	**5747**	-	-	3053	2519

	Bluefox FAST				
	A	B	C	D	E
park_7	**3648**	2981	2555	-	-
park_8	**3527**	668	-	358	-
park_9	**4060**	3061	-	-	2539
park_10	**4195**	-	2985	618	-
park_11	**4358**	-	2419	-	2689
park_12	**4733**	-	-	704	3040

	Shi-Tomasi				
	A	B	C	D	E
park_1	**4670**	541	532	-	-
park_2	**4369**	2769	-	2474	-
park_3	**5041**	1136	-	-	2536
park_4	**5194**	-	643	4211	-
park_5	**4302**	-	1499	-	2963
park_6	**4618**	-	-	3605	2887

	Shi-Tomasi				
	A	B	C	D	E
park_7	**3853**	3165	2911	-	-
park_8	**3527**	685	-	3070	-
park_9	**3878**	3114	-	-	2687
park_10	**4023**	-	3313	2419	-
park_11	**4305**	-	2673	-	4305
park_12	**4401**	-	-	1177	2985

	SIFT				
	A	B	C	D	E
park_1	**2551**	201	140	-	-
park_2	**2398**	1175	-	607	-
park_3	**2462**	397	-	-	791
park_4	**2319**	-	212	920	-
park_5	**2349**	-	540	-	893
park_6	**2364**	-	-	1058	782

	SIFT				
	A	B	C	D	E
park_7	**1943**	1491	1289	-	-
park_8	**1684**	341	-	182	-
park_9	**2065**	1488	-	-	1089
park_10	**2159**	-	1482	324	-
park_11	**2236**	-	1173	-	1222
park_12	**2710**	-	-	356	1356

Table 4.2: Results of the preliminary testing. The considered methods are abbreviated using the

Figure 4.3: Overview of the preliminary testing area.

As seen in Table 4.2, the built-in AEC and AGC performed better in all applications. Although that does not imply that it is the best control algorithm, it shows that in an environment saturated with information it outperforms the more advanced methods, such as the gradient-based methods, which showed a tendency to overexpose the scene, because it provided stronger gradients. However, that resulted in the removal of less prominent features in the image, such as sky or tree foliage.

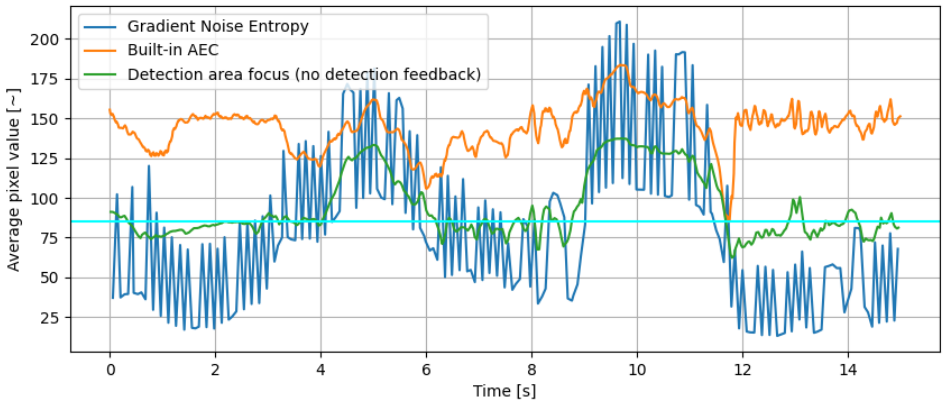


Figure 4.4: The average grayscale value from the park_12 dataset using different control algort. The light blue line marks the control value passed to the Built-in AEC and Detection area focus with no detection feedback algorithms.

The Gradient Noise Entropy control algorithm oscillated heavily as can be observed in Figure 4.4. It may be argued that these oscillations are caused by too high constants used in the simplex. The main reason for this oscillation is the discontinuity of the Nelder-Mead simplex, amplified by the adjustment for dynamic scenes.

Figure 4.5: The experimented MAV platform at the launch location equipped with three Bluefox cameras.

4.3 Application on drones

The results presented in the previous section provide an insight into the possible applications and the usability of implemented methods. The Gradient, noise, and entropy method was deemed not usable on drones due to its tendency to oscillate resulting in only three methods reaching satisfactory results for their application onboard an MAV.

Configuration

The experiment was conducted on an MAV provided by the Multi-robot Systems Group group [44], [45] and operating using the MRS UAV system [46]. The drone was equipped with a specialized frame designed for mounting the camera holders. The camera holder with the cameras was mounted on the frame, extending approximately 20 centimeters from the MAV axis and aiming directly forward. The mounting can be seen in Figure 4.5.

When mounted on the MAV, the four Basler cameras overwhelmed the bandwidth of the available USB ports, which caused random conflicts between transferred data packets. When the camera firmware detected such collision, it evaluated it as a critical error and proceeded to restart or shut down the camera. This interrupted the ongoing AEC and detection task. Because this behavior was consistent even with three cameras, Basler cameras were ruled out and the experiment onboard the MAV was conducted only using three Bluefox cameras.

The target was first placed flat on the ground. However, in combination with a low image resolution and the almost perpendicular position of the target relative to the camera's optical axis, the resulting detection was unreliable. To eliminate this problem, the target was erected at 45° angle, which greatly improved its visibility for the cameras, resulting in an improvement in the target detection. An example of scene perceived by the cameras onboard an MAV can be seen in Figure 4.6.

(a) Example from datasets drone_6 to drone_10, taken at sunset and with the target erected.

(b) Example from datasets drone_11 and drone_12, taken at sunrise and with the target erected.

Figure 4.6: Example of scenes captured by a camera running the Built-in AEC algorithm onboard a MAV.

The MAV started from the ground approximately 50 meters away from the target. It settled at a height in the range of 2 to 7 meters, and then slowly approached the target in a straight line, with occasional corrections, until it flew over it. Then it stopped and proceeded to return on the same trajectory without turning. The MAV then landed, and the whole experiment was repeated several times, with different control algorithms.

The experiment was conducted in the evening and once in the morning. These times of day offer a high range of luminosity values. The scenes captured by the onboard cameras running different algorithms are shown in Appendix C.

Results

From the evaluation metrics introduced in section 4.2.2, the FAST metric, due to its evaluation speed, has been used for the evaluation of data obtained during the experiment onboard an MAV. Additionally, the DETECTION metric is introduced. It is defined as the smallest area of a valid detection of the target captured by the camera. This metric was introduced to show methods performance at keeping the detection target better exposed, resulting in a longer detection range. The AprilTag target is of a size 10×10 pixels meaning the best possible result is 100. Lower values of the DETECTION metric are thus better.

Table 4.5 evaluates the number of frames at which detection occurred realtive to the number of total frames in a given dataset and the performance compared to the Built-in AEC algorithm described in %.

The Detection area focus method achieved the best result, however, because it was the most tested algorithm, comparing it with other control methods would produce a biased result. There were two cases where it performed unexpectedly. The first case was at dataset drone_12 where it produced the only result that was worse than the Built-in AEC and Detection area

(a) Photo of the MAV mid-flight approaching the target.

(b) Photo of the setup of the detection target.

(c) Close-up photo of the MAV mid-flight.

Figure 4.7: Photos of the MAV in action during the experiment and the target setup.

focus without detection feedback. The second case was at the dataset drone_10, where it performed overwhelmingly better than the Built-in AEC.

Figures 4.8a and 4.9a show an analysis of datasets drone_10 and drone_12, which includes the average grayscale value of an image over time. In the first case shown in Figure 4.8, the Detection area focus algorithm reacted to target detection by increasing the exposure time resulting in an overall brighter image and a better exposed target. However, it did not affect the detection quality, which performed worse than both built-in AEC control and Detection area focus without detection feedback.

A	Built-in AEC & AGC control
B	Gamma correction mapping([4])
C	Gradient ascent ([15])
D	Gradient Noise Entropy ([17])
E	Detection area focus
F	Detection area focus (no detection feedback)

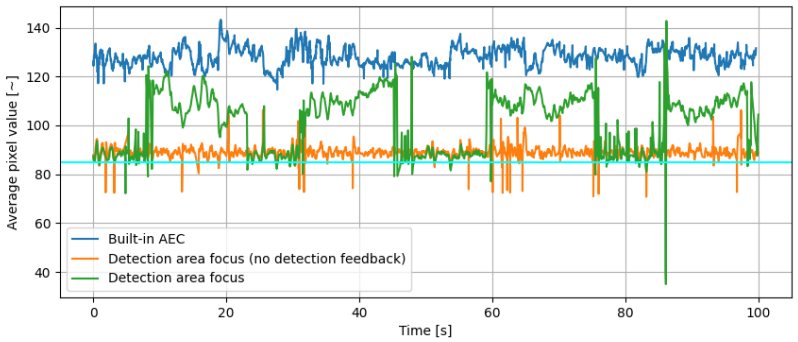
Table 4.3: The implemented and compared AEC and AGC methods.

	FAST						DETECT					
	A	B	C	D	E	F	A	B	C	D	E	F
drone_6	321	-	-	-	**340**	-	161	-	-	-	**139**	-
drone_7	549	-	-	-	**648**	-	155	-	-	-	**141**	-
drone_8	490	**711**	-	-	-	-	**163**	1325	-	-	-	-
drone_9	431	-	**2601**	-	-	-	**174**	-	NULL	-	-	-
drone_10	329	-	-	-	**341**		164	-	-	-	**122**	-
drone_11	873	-	-	-	**982**	870	155	-	-	-	144	**129**
drone_12	1078	-	-	-	**1360**	1113	155	-	-	-	160	**135**

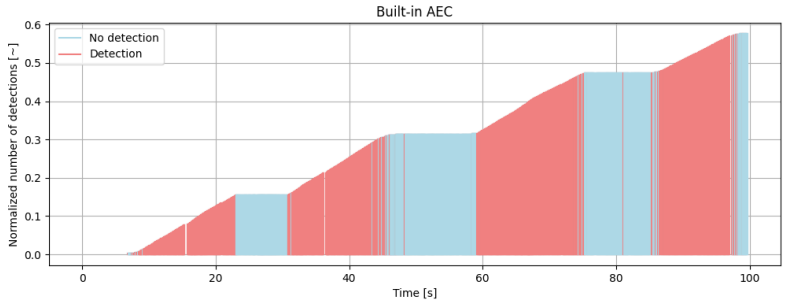
Table 4.4: Results of the onboard exposure control experiments. The best values are marked in red. The best values for a given dataset are marked with a bold font.

	Detection ratio						Comparison with the Built-in AEC (in %)					
	A	B	C	D	E	F	A	B	C	D	E	F
drone_6	0.211	-	-	-	**0.284**	-	100	-	-	-	**134.6**	-
drone_7	0.294	-	-	-	**0.299**	-	100	-	-	-	**101.7**	-
drone_8	0.471	0.140	-	-	-	-	**100**	29.7	-	-	-	-
drone_9	**0.384**	-	0.0	-	-	-	**100**	-	0.0	-	-	-
drone_10	0.162	-	-	-	**0.388**	-	100	-	-	-	**239.5**	-
drone_11	0.324	-	-	-	**0.364**	0.360	100	-	-	-	**112.3**	111.1
drone_12	0.493	-	-	-	0.491	**0.509**	100	-	-	-	99.6	**103.2**
Mean	0.334	0.140	0.0	-	0.365	**0.434**	100	29.7	0.0		**137.54**	107.15

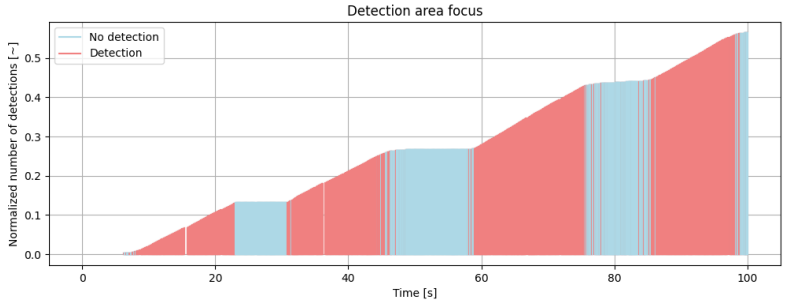
Table 4.5: Detection ratios and their deviation in percentage from Built-in Automatic Exposure Control.



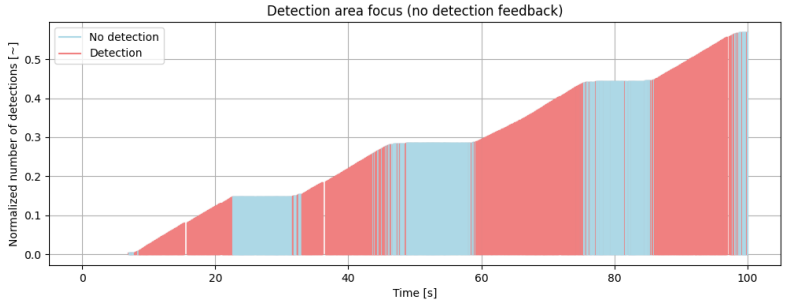
(a) The average grayscale values from the drone_12 dataset. The light blue line marks the desired control value passed to all three algorithms.



(b) Detection ration over time of the Built-in AEC in the drone_12 dataset.

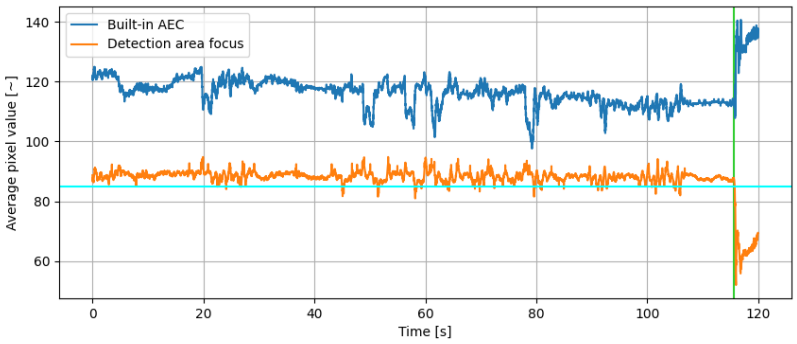


(c) Detection ration over time of the Detection area focus algorithm in the drone_12 dataset.

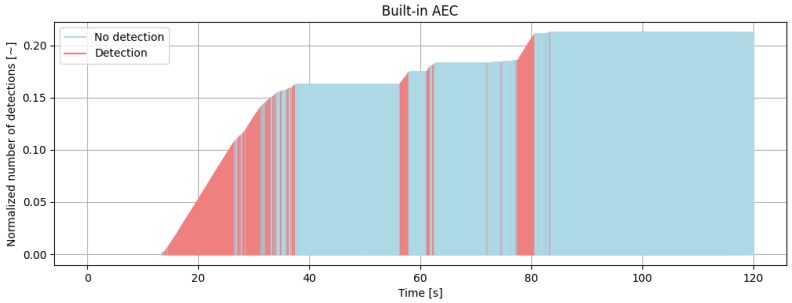


(d) Detection ration over time of the Detection area focus with no feedback from the detection task in the drone_12 dataset.

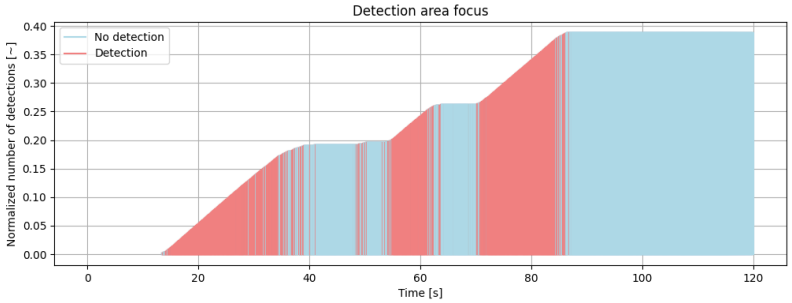
Figure 4.8: Case of the worst performance of the Detection area focus algorithm compared to Built-in AEC.



(a) The average grayscale values from the drone_10 dataset. The light blue line marks the desired control value passed to both algorithms. The green line marks the MAV landing.



(b) Detection ration over time of the Built-in AEC in the drone_10 dataset.



(c) Detection ration over time of the Detection area focus algorithm in the drone_10 dataset.

Figure 4.9: Case of the best performance of the Detection area focus algorithm compared to the Built-in AEC(drone_12 dataset).

The opposite happened in the second presented case, where the Detection area focus performed overwhelmingly better despite no clear effect on image brightness. This either implies that the target luminosity was similar to the average luminosity of the scene or the detection area focus algorithm did not receive any feedback from the detection task. Despite that, it demonstrates superior performance to the Built-in AEC control algorithm, as can be seen in the table 4.5.

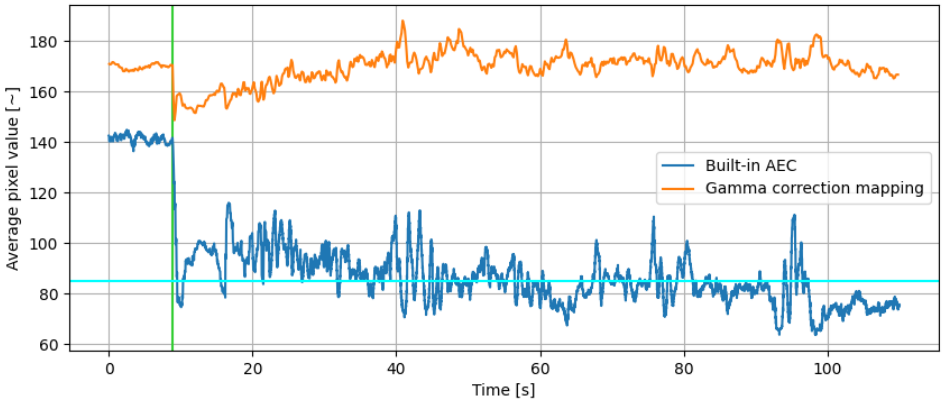


Figure 4.10: The average grayscale values from the drone_8 dataset. The light blue line marks the desired control value passed to the Built-in AEC algorithm. The green line marks the MAV takeoff.

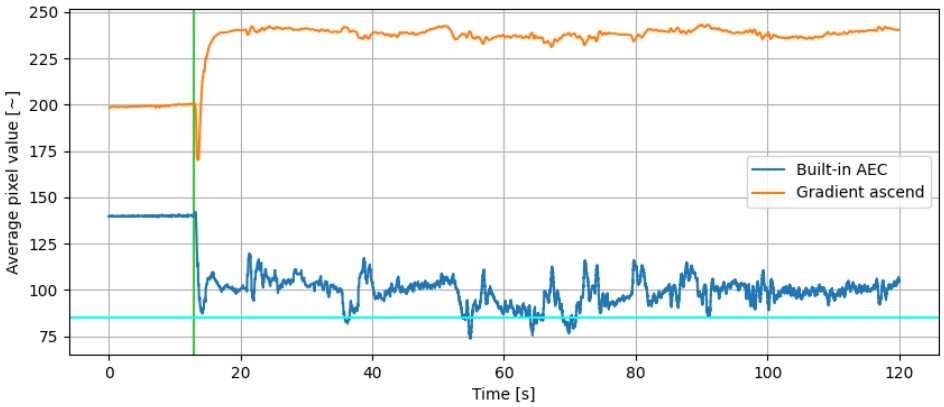
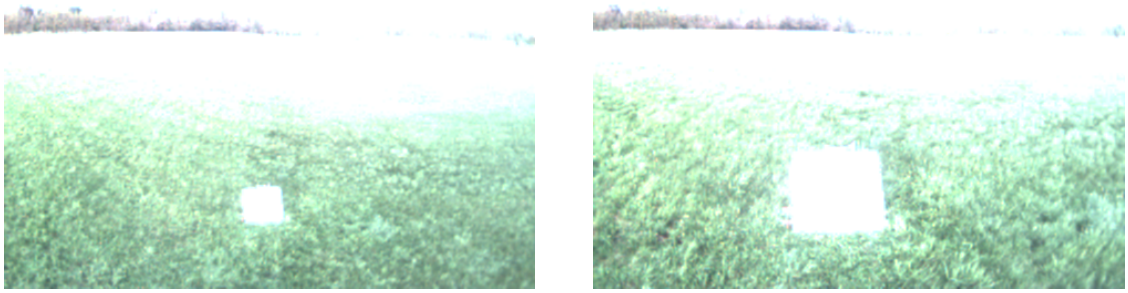


Figure 4.11: The average grayscale values from the drone_9 dataset. The light blue line marks the desired control value passed to the Built-in AEC algorithm. The green line marks the MAV takeoff.

The gradient-based methods yielded better results of the FAST metric than during the preliminary testing. This may be caused by areas of the image, such as the background forest, that had sharply defined gradients combined with a reduced feature-richness of the less prominent image regions. Despite that, their performance regarding the detection task was still worse than when the Built-in AEC was used.

Both used gradient methods suffered from overexposing the image, which hindered target detection. An extreme situation was encountered with the Gradient ascent method which



(a) Dataset drone_9, time = 46.7 s (b) Dataset drone_9, time = 94.6 s

Figure 4.12: Examples of scenes captured by the camera running the Gradient ascent algorithm.

did not manage to achieve a single detection. A high exposure time paired with the Sun's position caused the target to be strongly overexposed which made the AprilTag on the target appear completely white as can be seen in Figure 4.12.

4.4 Limitations

The implementation was faced with several forced and self-imposed limitations. The self-imposed limitations consisted of a minimum update rate of 20 Hz, to ensure competitiveness with Built-in AEC and high enough frame rate for possible subsequent task to run smoothly. These conditions were successfully achieved.

The forced limitation stemmed from the camera capabilities, such as the duration between setting an exposure time and camera applying the new exposure time. The considered cameras required three frames, meaning that after the evaluation of the current image and sending new values to the camera, the image at which the update took effect arrived three frames later. Thus, to ensure a minimal evaluation frame rate of 20 Hz, the minimum camera frame rate for the implemented control methods was set to 60 Hz. This was not a problem in the conducted experiments, as all of them were conducted in bright environment, but could pose problems in the future, if deployed in dimmer conditions.

Another limitation was the image resolution captured by the camera. As mentioned earlier, the image was cropped to limit the effect of the MAV rotors on the control algorithms. However, small resolution was could be the main reason the gradient based methods performed so poorly.

A significant amount of time was spent on resolving problems regarding the functionality of the cameras, such as missing or incorrect implementations in the API, and random disconnects, which were happening both with Bluefox and at a larger scale, Basler cameras. These problems were not foreseen and were discovered in the process of creating this work. Knowledge of these problems before the beginning of the implementation would be helpful.

5 Conclusion

In this thesis, several SOTA AEC methods were researched and implemented for an improvement in onboard detection task of a Micro Aerial Vehicle. These methods mostly employed control based on gradient information in the incoming images, paired with custom metrics to control the exposure time. Additionally, an AEC algorithm utilizing feedback from an onboard detection task was designed, implemented and compared with the other methods and the Built-in AEC as a baseline.

The gradient-based methods managed to control the camera exposure, but consistently produced overexposed images, severely limiting the detection. One explanation is an implementation mistake, however similar results were also achieved by [16], so it may be the flaw of the algorithm. These methods struggled or completely failed when the gradient information in the image was nearing zero.

A detection improvement was achieved by a Detection area focus algorithm with and without detection feedback. The improvement without detection feedback was too small to be conclusive, and more experiments are needed. The improvement with detection feedback is conclusive, as many tests were conducted resulting in an improvement of target detection by 137.5% on average compared to the Built-in AEC, when used onboard the MAV.

The experiments have shown that the high speed of convergence of the Built-in AEC algorithm can cause undesirable behavior, since its rapid adjustments caused a temporal overexposure or underexposure of the image, resulting potential temporal loss of the detected target. This could be negated by implementing a control algorithm with a slower response to changes. This might be one of the reasons why Detection area focus without detection feedback performed better.

The main conclusion of this work is that the introduction of a feedback loop from the onboard detection task results in improved detection performance compared to the Built-in AEC. The results of other methods performed worse than the Built-in AEC, however still managed to control the camera exposure. These methods could achieve better results in the future if the evaluation metrics are improved.

5.1 Future improvements & suggestions

One of the problems this implementation did not solve was the elimination of method parameter adjustment for specific environment. This was planned to be achieved by mapping parameter values to a scene radiance, similarly to photometric response function introduced in Section 3.1.2, which could, in theory, cover most scenarios.

Considering new method implementations, instead of targeting a wide variety of tasks, the new methods should focus more on a specified area of tasks, and then let the user pick the algorithm that is the best suited for their task. This means that the CNN-based algorithms, which were rejected due to their importability, could play a significant role here.

Through the experiments, it was demonstrated that it is not necessary to incorporate complex image evaluation techniques to reach an improvement of the onboard detection task. A simple and fast solution can perform better than the Built-in AEC. Future implementations should not focus on covering a whole range of scenes ranging from direct sunlight to dark

rooms, and instead limit themselves to a specific set of scenes, such as outdoor scenes during daylight, to make fine-tuning control methods easier.

6 References

[1] N. Sampat, S. Venkataraman, T. Yeh, and R. L. Kremens, "System implications of implementing auto-exposure on consumer digital cameras," in *Sensors, Cameras, and Applications for Digital Photography*, N. Sampat and T. Yeh, Eds., International Society for Optics and Photonics, vol. 3650, SPIE, 1999, pp. 100 –107. DOI: 10.1117/12.342854.

[2] J. Kim, Y. Cho, and A. Kim, "Proactive camera attribute control using bayesian optimization for illumination-resilient visual navigation," *IEEE Transactions on Robotics*, vol. 36, pp. 1256–1271, 2020.

[3] ——, "Exposure control using bayesian optimization based on entropy weighted image gradient," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 857–864, 2018.

[4] I. Shim, T.-H. Oh, J.-Y. Lee, J. Choi, D.-G. Choi, and I. S. Kweon, "Gradient-based camera exposure control for outdoor mobile platforms," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 6, pp. 1569–1583, 2019. DOI: 10.1109/TCSVT.2018.2846292.

[5] H. Lu, H. Zhang, S. Yang, and Z. Zheng, "Camera parameters auto-adjusting technique for robust robot vision," in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 1518–1523. DOI: 10.1109/ROBOT.2010.5509978.

[6] J. Tomasi, B. Wagstaff, S. L. Waslander, and J. Kelly, "Learned camera gain and exposure control for improved visual feature detection and matching," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2028–2035, 2021. DOI: 10.1109/LRA.2021.3058909.

[7] Y. Wang, H. Chen, S. Zhang, and W. Lu, "Automated camera-exposure control for robust localization in varying illumination environments," *Autonomous Robots*, vol. 46, pp. 1–20, Apr. 2022. DOI: 10.1007/s10514-022-10036-x.

[8] N. Nourani-Vatani, J. Roberts, and T. Centre, "Automatic camera exposure control," Jan. 2007.

[9] T. Jiang, K.-D. Kuhnert, D. Nguyen, and L. Kuhnert, "Multiple templates auto exposure control based on luminance histogram for onboard camera," in *2011 IEEE International Conference on Computer Science and Automation Engineering*, vol. 3, 2011, pp. 237–241. DOI: 10.1109/CSAE.2011.5952672.

[10] J. Torres and J. M. Menéndez, "Optimal camera exposure for video surveillance systems by predictive control of shutter speed, aperture, and gain," vol. 9400, Feb. 2015. DOI: 10.1117/12.2083182.

[11] W.-C. Kao, L.-W. Cheng, C.-Y. Chien, and W.-K. Lin, "Robust brightness measurement and exposure control in real-time video recording," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 4, pp. 1206–1216, 2011. DOI: 10.1109/TIM.2010.2087835.

[12] S. Park, G. Kim, and J. Jeon, "The method of auto exposure control for low-end digital camera," in *2009 11th International Conference on Advanced Communication Technology*, vol. 03, 2009, pp. 1712–1714.

[13] T. Gao, L. Zheng, W. Xu, Y. Piao, R. Feng, X. Chen, and T. Zhou, "An automatic exposure method of plane array remote sensing image based on two-dimensional entropy," *Sensors*, vol. 21, p. 3306, May 2021. DOI: 10.3390/s21103306.

[14] I. Shim, J.-Y. Lee, and I. S. Kweon, "Auto-adjusting camera exposure for outdoor robotics using gradient information," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 1011–1017. DOI: 10.1109/IROS.2014.6942682.

[15] Z. Zhang, C. Forster, and D. Scaramuzza, "Active exposure control for robust visual odometry in hdr environments," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 3894–3901. DOI: 10.1109/ICRA.2017.7989449.

[16] I. Mehta, M. Tang, and T. D. Barfoot, "Gradient-based auto-exposure control applied to a self-driving car," in *2020 17th Conference on Computer and Robot Vision (CRV)*, 2020, pp. 166–173. DOI: 10.1109/CRV50864.2020.00030.

[17] U. Shin, J. Park, G. Shim, F. Rameau, and I. Kweon, "Camera exposure control for robust robot vision with noise-aware image quality assessment," Jul. 2019. DOI: 10.1109/IROS40897.2019.8968590.

[18] Y. Su and C.-C. Jay Kuo, "Fast and robust camera's auto exposure control using convex or concave model," in *2015 IEEE International Conference on Consumer Electronics (ICCE)*, 2015, pp. 13–14. DOI: 10.1109/ICCE.2015.7066300.

[19] H. Yang, B. Wang, N. Vesdapunt, M. Guo, and S. B. Kang, "Personalized exposure control using adaptive metering and reinforcement learning," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 10, pp. 2953–2968, 2019. DOI: 10.1109/TVCG.2018.2865555.

[20] K. Lee, U. Shin, and B.-U. Lee, *Learning to control camera exposure via reinforcement learning*, 2024. arXiv: 2404.01636 [cs.CV].

[21] E. Onzon, F. Mannan, and F. Heide, "Neural auto-exposure for high-dynamic range object detection," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 7706–7716. DOI: 10.1109/CVPR46437.2021.00762.

[22] M. Bigas, E. Cabruja, J. Forest, and J. Salvi, "Review of cmos image sensors," *Microelectronics Journal*, vol. 37, no. 5, pp. 433–451, 2006, ISSN: 0026-2692. DOI: https://doi.org/10.1016/j.mejo.2005.07.002.

[23] M. El-Desouki, M. Jamal Deen, Q. Fang, L. Liu, F. Tse, and D. Armstrong, "Cmos image sensors for high speed applications," *Sensors*, vol. 9, no. 1, pp. 430–444, 2009, ISSN: 1424-8220. DOI: 10.3390/s90100430.

[24] M. Schöberl, S. Fößel, H. Bloss, and A. Kaup, "Modeling of image shutters and motion blur in analog and digital camera systems," in *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 3457–3460. DOI: 10.1109/ICIP.2009.5413842.

[25] Y.-C. Liu, W.-H. Chan, and Y.-Q. Chen, "Automatic white balance for digital still camera," *IEEE Transactions on Consumer Electronics*, vol. 41, no. 3, pp. 460–466, 1995. DOI: 10.1109/30.468045.

[26] C.-C. Weng, H. Chen, and C.-S. Fuh, "A novel automatic white balance method for digital still cameras," in *2005 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2005, 3801–3804 Vol. 4. DOI: 10.1109/ISCAS.2005.1465458.

[27] T. Oskam, A. Hornung, R. W. Sumner, and M. Gross, "Fast and stable color balancing for images and augmented reality," in *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, 2012, pp. 49–56. DOI: 10.1109/3DIMPVT.2012.36.

[28] H.-C. Lee, *Introduction to Color Imaging Science*. Cambridge University Press, 2005.

[29] Y. Wan and Q. Xie, "A novel framework for optimal rgb to grayscale image conversion," in *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 02, 2016, pp. 345–348. DOI: 10.1109/IHMSC.2016.201.

[30] S. Xie, A. A. Prouza, and A. Theuwissen, "A cmos-imager-pixel-based temperature sensor for dark current compensation," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 2, pp. 255–259, 2020. DOI: 10.1109/TCSII.2019.2914588.

[31] I. Takayanagi, J. Nakamura, E. Fossum, K. Nagashima, T. Kunihoro, and H. Yurimoto, "Dark current reduction in stacked-type cmos-aps for charged particle imaging," *IEEE Transactions on Electron Devices*, vol. 50, no. 1, pp. 70–76, 2003. DOI: 10.1109/TED.2002.806963.

[32] R. Peters, "A new algorithm for image noise reduction using mathematical morphology," *IEEE Transactions on Image Processing*, vol. 4, no. 5, pp. 554–568, 1995. DOI: 10.1109/83.382491.

[33] Y. Zhu and C. Huang, "An improved median filtering algorithm for image noise reduction," *Physics Procedia*, vol. 25, pp. 609–616, 2012, International Conference on Solid State Devices and Materials Science, April 1-2, 2012, Macao, ISSN: 1875-3892. DOI: https://doi.org/10.1016/j.phpro.2012.03.133.

[34] D. Van De Ville, M. Nachtegael, D. Van der Weken, E. Kerre, W. Philips, and I. Lemahieu, "Noise reduction by fuzzy image filtering," *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 4, pp. 429–436, 2003. DOI: 10.1109/TFUZZ.2003.814830.

[35] M. Özkan, I. Sezan, and A. Tekalp, "Adaptive motion-compensated filtering of noisy image sequences," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 3, pp. 277 –290, Sep. 1993. DOI: 10.1109/76.257217.

[36] K. Miyata and A. Taguchi, "Spatio-temporal separable data-dependent weighted average filtering for restoration of the image sequences," in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, 2002, pp. IV–3696–IV–3699. DOI: 10.1109/ICASSP.2002.5745458.

[37] H. Malm, M. Oskarsson, E. Warrant, P. Clarberg, J. Hasselgren, and C. Lejdfors, "Adaptive enhancement and noise reduction in very low light-level video," in *2007 IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8. DOI: 10.1109/ICCV.2007.4409007.

[38] M. Grossberg and S. Nayar, "Modeling the space of camera response functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 10, pp. 1272–1282, 2004. DOI: 10.1109/TPAMI.2004.88.

[39] J. A. Nelder and R. Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, Jan. 1965, ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308.

[40] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the nelder-mead simplex method in low dimensions," *SIAM J. Optim.*, vol. 9, pp. 112–147, 1998.

[41] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 2, 2005, 1508–1515 Vol. 2. DOI: 10.1109/ICCV.2005.104.

[42] J. Shi and Tomasi, "Good features to track," in *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600. DOI: 10.1109/CVPR.1994.323794.

[43] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, 1999, 1150–1157 vol.2. DOI: 10.1109/ICCV.1999.790410.

[44] D. Hert, T. Baca, P. Petracek, V. Kratky, R. Penicka, V. Spurny, M. Petrlik, M. Vrba, D. Zaitlik, P. Stoudek, V. Walter, P. Stepan, J. Horyna, V. Pritzl, M. Sramek, A. Ahmad, G. Silano, D. B. Licea, P. Stibinger, T. Nascimento, and M. Saska, "Mrs drone: A modular platform for real-world deployment of aerial multi-robot systems," *Journal of Intelligent and Robotic Systems*, vol. 108, no. 4, 2023, ISSN: 1573-0409. DOI: 10.1007/s10846-023-01879-2.

[45] D. Hert, T. Baca, P. Petracek, V. Kratky, V. Spurny, M. Petrlik, M. Vrba, D. Zaitlik, P. Stoudek, V. Walter, P. Stepan, J. Horyna, V. Pritzl, G. Silano, D. Bonilla Licea, P. Stibinger, R. Penicka, T. Nascimento, and M. Saska, "Mrs modular uav hardware platforms for supporting research in real-world outdoor and indoor environments," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2022. DOI: 10.1109/icuas54217.2022.9836083.

[46] T. Báča, M. Petrlík, M. Vrba, V. Spurný, R. Pěnička, D. Hert, and M. Saska, "The mrs uav system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles," *Journal of Intelligent and Robotic Systems*, vol. 102, May 2021. DOI: 10.1007/s10846-021-01383-5.

6.1 Used Generative AI

List of software tools that affected the quality of this work.

ChatGPT:

- OpenAI. (n.d.). *ChatGPT (Version 4.0) [Software]*. Retrieved from https://www.openai.com/gpt
  Usage: Generation of simple python scripts and illustrative examples of the use of OpenCV C++ functions. Assisted with an explanation of functionality of C++ and OpenCV.

QuillBot:

- QuillBot. (n.d.). *QuillBot [Software]*. Retrieved from https://quillbot.com/
  Usage: Proofreading and grammar assistance paired with structural improvement of sentences.

Grammarly:

- Grammarly. (n.d.). *Grammarly [Software]*. Retrieved from https://app.grammarly.com/
  Usage: Proofreading.

A Implemented method parameters

Gamma correction mapping	
δ	0.8
λ	1000
d	0.5
K_{p}	0.1

Gradient ascent	
γ	0.8
k	7
p	0.00001

Gradient, noise and entropy	
δ	0.1
λ	1000
K_{g}	2
K_{e}	0.125
τ_{h}	240
τ_{l}	15
p	0.1
α	0.4
β	0.4
ϵ	0.4
N_{C}	100
ϵ	1.7
$\alpha_{\mathrm{simplex}}$	0.2
$\gamma_{\mathrm{simplex}}$	1.2
ρ_{simplex}	0.2
$\sigma_{\mathrm{simplex}}$	0.2

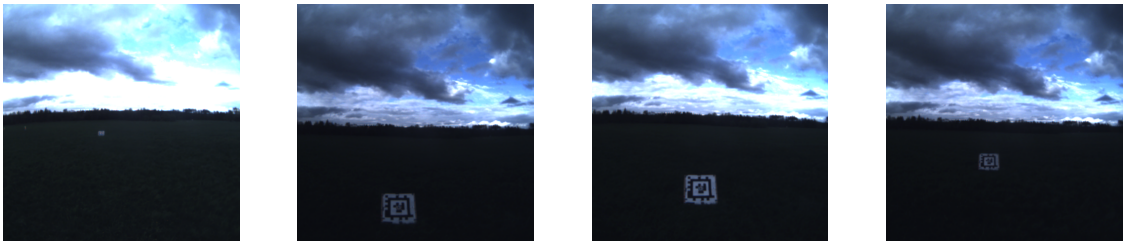
Detection area focus	
Control value	85
γ	0.4

B Algorithms

Algorithm 1 Nelder-Mead Simplex Algorithm

Init simplex
  $\mathbf{x}_0 \leftarrow [\text{ExposureTime0}, \text{Gain0}]$
  $J_{\text{x0}} \leftarrow \text{mean}(\text{Image})$
  $h \leftarrow \begin{cases} -\epsilon^{-1}(J_{\text{x0}}/255) & \text{if } J_{\text{x0}} \geq 128 \\ \epsilon^{-1}(1 - J_{\text{x0}}/255) & \text{if } J_{\text{x0}} < 128 \end{cases}$
  $\mathbf{x}_{\text{i}} \leftarrow \mathbf{x}_0 + h(\mathbf{I}_2 \cdot \mathbf{x}_0)$
  Evaluate $f(I)$ for $\mathbf{x}_{\text{i}}$, where $\text{i} = 1, 2$
Sort
  Sort f_{i} in descending order
Compute centroid
  $\mathbf{x}_{\text{c}} \leftarrow \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_{\text{i}}$
Reflect
  $\mathbf{x}_{\text{r}} \leftarrow \mathbf{x}_{\text{c}} + \alpha(\mathbf{x}_{\text{c}} - \mathbf{x}_2)$
Replace
  if $f_{\text{r}} < f_1$ **then**
    Replace $\mathbf{x}_2$ with $\mathbf{x}_r$
  end if
Expand
  $\mathbf{x}_{\text{e}} \leftarrow \mathbf{x}_{\text{c}} + \gamma(\mathbf{x}_{\text{r}} - \mathbf{x}_c)$
  if $f_{\text{e}} < f_{\text{r}}$ **then**
    Replace $\mathbf{x}_2$ with $\mathbf{x}_e$
  else
    Replace $\mathbf{x}_2$ with $\mathbf{x}_r$
  end if
Contract
  if $f_r < f_{n+1}$ **then**
    Outside contraction: $x_{oc} = x_c + \rho(x_r - x_c)$
    if $f_{oc} \leq f_r$ **then**
      Replace x_2 with x_{oc}
    else
      Shrink towards the best vertex: $x_{\text{i}} = x_0 + \sigma(x_{\text{i}} - x_0)$ for $i = 2, 3$
    end if
  else
    Inside contraction: $x_{ic} = x_c - \rho(x_c - x_2)$
    if $f_{ic} < f_{n+1}$ **then**
      Replace $x^{(n+1)}$ with x_{ic}
    else
      Shrink towards the best vertex: $x_{\text{i}} = x_0 + \sigma(x_{\text{i}} - x_0)$ for $i = 2, 3$
    end if
  end if

C Scene Examples

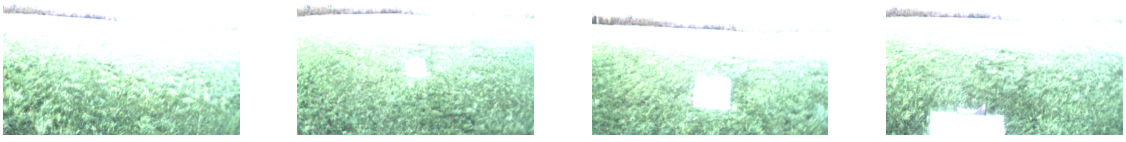


(a) (b) (c) (d)

Figure C.1: Scene examples from dataset drone_9 captured by camera using the Built-in AEC.

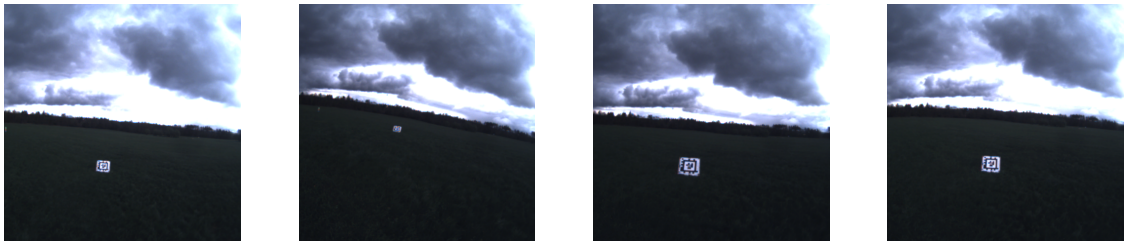
(a) (b) (c) (d)

Figure C.2: Scene examples from dataset drone_8 captured by camera using the Gamma correction mapping method.



(a) (b) (c) (d)

Figure C.3: Scene examples from dataset drone_9 captured by camera using the Gradient ascent method.

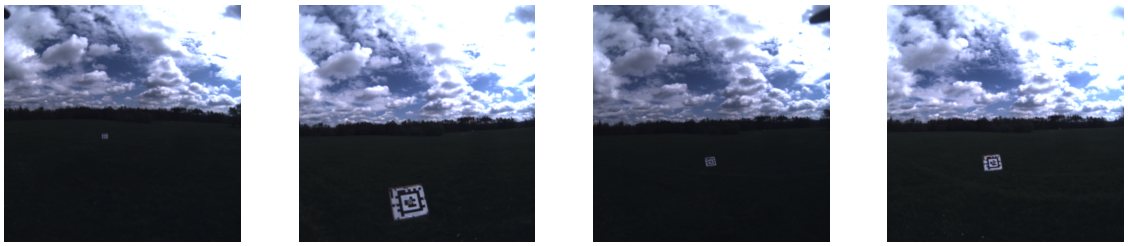


(a) (b) (c) (d)

Figure C.4: Scene examples from dataset drone_6 captured by camera using the Detection focus method. The target detection was present at (b) and (d).

(a) (b) (c) (d)

Figure C.5: Scene examples from dataset drone_12 captured by camera using the Built-in AEC.



(a) (b) (c) (d)

Figure C.6: Scene examples from dataset drone_12 captured by camera using the Detection focus method. The target detection was present at (b).