UiT The Arctic University of Norway

Faculty of Science and Technology
Department of Computer Science

DeepRoom: A Deep Learning Rating System for Photography

Victoria Kumetz Lillegård Lintvedt

INF-3981 Master's Thesis in Computer Science - June 2023

“It’s basically Pokémon Snap”
–Jørgen Gauthun Frøseth

“IT’S ART!”
–Ingvild Alvarstein

Abstract

This thesis explores integrating deep learning techniques into photography, aiming to automate the identification of good images within large datasets. The primary focus is developing a deep learning-based system called *DeepRoom* that rates and evaluates photographs based on photography-specific technical criteria. To accomplish this, the research methodology encompasses qualitative research alongside developing a system prototype. A section overviews deep learning, photography, and related work and emphasizes its relevance to the research objectives. Implementation details include describing development tools and processes employed to construct the deep learning models and curate the dataset. These models' performance is assessed in the following evaluation phase, and a comparative analysis is conducted against existing software solutions. Encouraging results are observed, particularly in object detection and exposure classification, while identifying areas for improvement, such as refining the blurry and skewed horizon models. In conclusion, this research highlights the contributions of DeepRoom and proposes future work, including dataset expansion and model refinement, to enhance its capabilities further.

Acknowledgements

First and foremost, I want to thank my primary supervisor Anne Håkansson for all the help during the past year, from helping me develop the concept to naming the system. I also want to thank my co-supervisor Aya Saad for the valuable insight and knowledge you brought to the project. Thank you both for the invaluable feedback, support, and pleasant conversations.

Thank you to all my fellow students at UiT for making a fun work environment and for all the pleasant conversations when we procrastinated about doing actual work.

Thank you to all of my close friends for providing a distraction when needed and for lending an ear when I was complaining about my studies these past five years. You know who you are, and I appreciate you all.

My last thanks go to my family who has *always* believed in me and supported me. A special thank you goes to my dad, who has always provided immense help and support throughout my studies. Thank you for the countless hours you have put in to ensure that I do my best, for the brainstorming and discussions, and for the incredibly helpful insight, knowledge, and feedback. I could not have done this without you.

Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Network
AP	Average Precision
CLI	Command-line Interface
CNN	Convolutional Neural Network
DL	Deep Learning
GUI	Graphical User Interface
JPEG	Joint Photographic Experts Group
mAP	Mean Average Precision
YOLO	You Only Look Once

1

Introduction

Artificial intelligence (AI) is, in its simplest form, described as everything that involves a computer solving a task similar to a human [1]. AI has become increasingly popular in the 21st century [2], and adoption of AI technology has more than doubled in the last five years, according to a survey performed by McKinsey & Company in 2022 [3]. To name a few more extensive adoption of AI in recent years, there is OpenAI's chatbot *ChatGPT* [4] that was released in November 2022 and gained an estimated 100 million users by January 2023, making it the fastest-growing consumer user base in the history of the internet [5]. As well as Tesla [6], who are using neural networks to create autopilot systems for their cars [7]. This is achieved by using real-time input from cameras in the car to analyze the environment using deep neural networks, providing a system that gives the car immediate feedback on the surrounding environment being able to pick up on occurring hazards and avoiding them [7].

Artificial intelligence has become an integral part of people's daily life, with AI applications used in several industries, such as technology, social media, marketing, entertainment, and banking [8]. Although artificial intelligence has only gained much traction with the general public in recent years, it has been a concept since the 1940s, considered "The Birth of AI" [9]. However, the term was not coined until 1956 by one of the founders of AI; John McCarthy [9]. AI is a broad term that envelops subsets such as *machine learning*, a field that focuses on making a machine learn [10], *artificial neural networks*, which draws inspiration from biological neural networks and aims to mimic how a human brain works [11], and *deep learning* (DL), which seeks to detect data patterns using deep neural networks [12]. These concepts were presented in the 1940s and 1950s [9].

Computer vision, in its essence, aims to replicate human vision [13] through mediums such as digital imaging and videos to derive meaningful information from visual inputs [14]. In some ways, computer vision can be traced back to 1957 when the first digital image was created [15, 16, 17], but was first presented as a concept in 1963 by Lawrence Roberts [17]. Computer vision uses AI technology such as deep learning and neural networks to create systems that somewhat replicate human vision [18] and can be used for various applications. With its wide range of applications, computer vision enhances various aspects of daily life, including automation, faster analysis, and enhanced security. It is integrated into numerous technologies, such as facial recognition on mobile phones, image search, and gaming [19].

Our everyday lives are impacted by artificial intelligence [20], and it has arguably made our lives easier. Gone are the days of developing negatives and expensive camera equipment with the introduction of smartphones and digital cameras. Nowadays, everyone with a smartphone, which 86.29% of the world population owns [21], has a camera at their fingertips. Photography and digital images have become a part of many people's everyday life, especially with social media platforms such as Instagram [22] and Facebook [23], which encourage their users to post their images [24]. With photography being so widely accessible, it is easy for people to accumulate large amounts of images that take up storage on their phones, computers, hard drives, or servers. Also, professional photographers can shoot about 1500 photos during a photoshoot.

1.1 Background and Motivation

Photographers who actively capture images over an extended period often face two significant challenges: storage limitations and information overload. Accumulating a vast library of photographs poses storage concerns, requiring effective management strategies. Moreover, sifting through thousands of images and eliminating duplicates or unnecessary files results in information overload. Coping with vast amounts of data has become increasingly challenging in the digital age [25].

During a typical photo shoot, it is common to accumulate images ranging from 1000 to 2000. Consequently, manually reviewing and selecting the best photographs for post-production takes time. To address these challenges, this research investigates the possibility of automating the process of identifying the best images in a given dataset using deep learning.

1.2 Problem

Deep learning and computer vision are widely used in product development and research. Applying deep learning to imagery can be a significant resource to help people in many fields. Within the research community, most DL research has been focused on creating models to help with medical evaluations, such as finding tumors or diseases in imagery. In this scenario, the DL model will focus on finding a specific thing in the image, disregarding the rest.

Photographs are complex, and there is more than one thing to consider when evaluating the image. Several aspects of the image need to be looked at and judged accordingly. These aspects will also vary depending on whether the image contains an animate subject. This requires the deep learning models to be quite complex and pick up on different parts of the photography to make a calculated decision on whether it should be deemed good. Additionally, what is considered good photography is a highly subjective manner. Creating a system that can determine a good photograph from a bad one is complex and will likely only align with some consumers' perspectives.

This thesis explores how deep learning can be used in imagery. The type of imagery is focused on photography, as evaluating photographs is a complex task involving multiple aspects. It investigates the design and implementation of a system to determine whether a photograph is good based on photography-specific technical criteria. The focus is on creating deep-learning models that adequately rate and evaluate photographs.

1.3 Methodology

The methodology is integral to working on a research project to ensure accurate and well-founded results [26]. To create a solidified research problem, a literary study is often performed. This helps gain knowledge surrounding the thesis subject and create a solid background before conducting the research. Furthermore, developing a strategy for the thesis goals and results to be reached is important. Various research methods can help build up the strategy, which will help create reliable, correct, and valid results [26, 27].

Methods refer to the techniques, procedures, or means employed to perform a particular task in a well-organized, systematic, and rational way [27, 28]. When writing a thesis, the first decision is whether to use qualitative or quantitative research methods. The former focuses on using existing research within the field to deepen the knowledge surrounding the topic and create theories, products, and inventions utilizing that knowledge. The latter focuses on testing and understanding how or whether something works through experimentation and testing to conclude while laws, theories, and principles are formed [26, 27]. Qualitative research aims to create a hypothesis, while quantitative research seeks to test whether a hypothesis is correct [26].

For this project, a *qualitative research* approach is chosen as the goal is to develop a system prototype instead of confirming or falsifying a theory. Through this thesis, a hypothesis is proposed based on research within the field of artificial intelligence on whether deep learning can be used to create a system to determine a good photograph. This hypothesis is tested through the implementation and experimentation of the system.

1.4 Contribution

DeepRoom significantly contributes to the research field by developing a deep learning-based rating system that assesses various aspects of image complexity, including exposure and blurriness. This automated system saves time and provides a consistent and standardized method for rating photographs. DeepRoom introduces multiple specialized deep learning models for comprehensive and accurate evaluations.

Moreover, this thesis explores deep learning principles to create a system that assesses the composition of photographs from a technical perspective. DeepRoom considers both animate and inanimate images. This broader approach distinguishes DeepRoom from traditional methods focusing solely on specific subjects within the photo.

Additionally, the practical implementation of DeepRoom utilizes powerful image recognition techniques, making it versatile for various applications, including satellite or medical imaging analysis.

Overall, DeepRoom's contribution lies in its innovative approach to objectively assessing photographs, its comprehensive evaluation of multiple composition aspects, and its potential for broader applications beyond traditional photography evaluation.

1.5 Delimination

This thesis does not prioritize the development of a comprehensive system with a functional command-line interface or graphical user interface. Instead, the primary focus is on developing the core functionality, which involves creating and refining the deep learning models for the prototype system. The objective is to demonstrate the feasibility of rating photographs using these models. By emphasizing the functionality of the deep learning models, this research aims to establish a solid foundation for the rating system's implementation and showcase its potential effectiveness.

1.6 Outline

In the subsequent chapters, the thesis will follow this structure:

Chapter 2: Background provides an exploration of the technical background underlying this thesis. It encompasses essential deep learning concepts relevant to the research, an examination of photography terminology, and an overview of the related work in the field.

Chapter 3: Methodology outlines the methodology employed in this research project, including the selected methods, their theoretical foundations, dataset, and implementation details.

Chapter 4: Design introduces the system's design and architecture, outlines requirements, and defines the criteria for rating photographs.

Chapter 5: Implementation details a comprehensive overview of the system's implementation, including the development tools used, the construction of deep learning models, and details about the dataset.

Chapter 6: Evaluation evaluates the performance of the deep learning models, and a comparison is made between the system and existing software.

Chapter 7: Discussion highlights and discusses the observed challenges encountered during the research.

Chapter 8: Conclusions and Future Work provides a summary and conclusion of the thesis, along with suggestions for future work.

2

Background

This chapter will cover the technical background for this thesis. It will give a general and relevant description of *deep learning* and *computer vision* as well as describe the relevant technicality behind the composure of a photograph. This section will also look at and analyze existing work related to this thesis.

2.1 Deep Learning

Artificial intelligence is a broad term that covers a large field of subjects. The terminology covers everything that makes a machine solve tasks similar to a human being [1]. Within AI as a term, there are several subsets, the main ones being *machine learning (ML)*, *artificial neural networks (ANN)*, and *deep learning (DL)*. Each of these subsets correlates, with ANN being a subset of ML and DL being a subset of ANN, as visualized in Figure 2.1. Machine learning enables machines or computers to acquire knowledge through learning capabilities without explicitly instructing the computer on what actions to take [10]. Artificial neural networks, inspired by biological neural networks, are constructed using algorithms that mimic the human brain [11].

Deep learning is a machine learning technique that utilizes deep neural networks (DNN) with three or more hidden layers to detect data patterns [12]. It aims to optimize the output of a neural network by adding more hidden layers. Deep learning is specifically designed for processing unstructured data, such as raw images, without the need for extensive pre-processing [29].

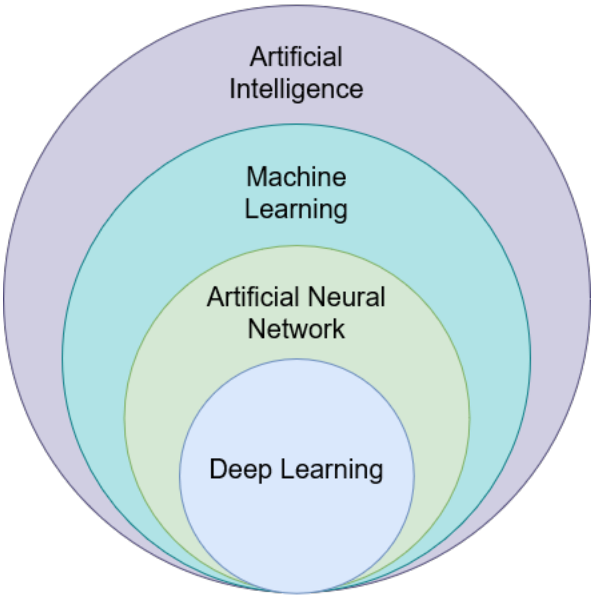


Figure 2.1: Visualisation of the subsets of artificial intelligence and how they correlate to each other [30].

Artificial Neural networks consist of layers, including an input layer, an output layer, and one or more hidden layers. Each layer comprises artificial neurons (nodes) that mimic biological neurons. These neurons are interconnected across layers and possess weighted values and associated biases [11].

Weights and biases are fundamental components of neural networks, influencing the strength of connections between neurons in different layers [31]. Biases are constant values added to the input and weights, shifting the activation function and facilitating the artificial neural network (ANN) training. The activation function determines whether an input is activated and forwarded to the next layer [32]. In this process, the neural network computes the activation function equation (2.1) while considering weighted connections between neurons. The weights and biases are learnable attributes that evolve during the network's training [31].

$$Y = \sum (weight * input) + bias \tag{2.1}$$

Normalization is a crucial consideration when training neural networks as it facilitates model training by adjusting the input data. If the interval between input values is excessively large, it can significantly increase training time and make it challenging for neurons to activate. This occurs because weights, typically small numbers, are the sole means of adjusting the input. To address this, normalization scales the input values to a more appropriate interval, ultimately enhancing validation accuracy [33].

Backpropagation is a fast and widely used algorithm for training feedforward neural networks. It involves calculating the error rate between the network's output and the target output, allowing the algorithm to adjust the weights individually for each neuron. The algorithm works by going backward through the neural network and modifying the weights in each layer based on the error rate. When the forward pass is repeated, the weights are adjusted again for each layer, ensuring minimal changes. This efficient learning approach considers the error rate and enables precise weight adjustments during training, making it faster than traditional methods. Unlike a regular feedforward neural network, backpropagation accounts for the error rate from previous executions, enhancing the learning process [34].

2.1.1 Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNN) are a type of neural network that processes data through convolution, mimicking how a human eye captures an image. CNNs excel at computationally intensive tasks like image processing, as they identify and refine patterns or features within images through successive convolutional layers. This enables tasks like image recognition. CNNs are more cost-effective than other machine learning techniques because they extract essential data from the dataset, processing what is necessary, which saves time and resources [35].

CNNs are commonly used for tasks such as image classification, object detection, and image segmentation [36]. As depicted in Figure 2.2, a CNN consists of an input layer followed by convolutional layers. Within the convolutional layers, patterns are recognized and extracted from the data, which then flows into subsequent layers. The resulting data from the convolutional layers are eventually fed into a traditional feed-forward neural network for final processing and output generation [35].

Figure 2.2 provides a visual representation of a CNN, where the yellow neurons represent the input layer, the purple neurons symbolize the convolutional layers, the green neurons depict the hidden layers, and the red neurons signify the output layer.

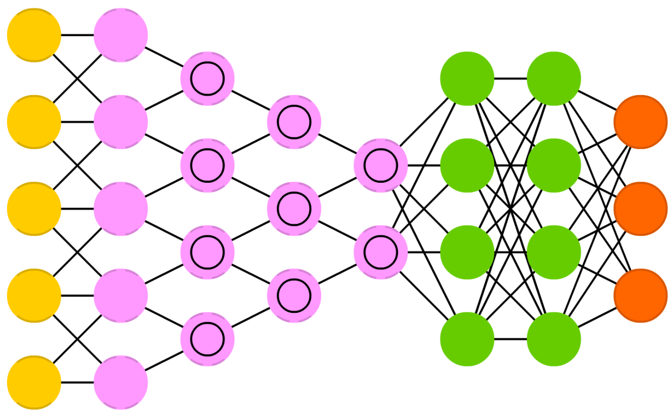


Figure 2.2: Illustration of a CNN [37].

2.1.2 Deep Learning for Images

Deep learning techniques have played a crucial role in advancing computer vision, which focuses on replicating aspects of human vision to extract meaningful information from visual inputs, such as digital images and videos [18, 13, 14]. Deep learning is employed in computer vision tasks such as image classification, object detection, and image segmentation [38].

Image classification involves labeling and categorizing a whole image, while image localization can identify and label a specific object within an image [38]. Object detection extends this to handle multiple objects, such as labeling cats and dogs separately [39]. Image segmentation divides an image into segments or clusters to classify and label objects more precisely by creating masks around them. This approach provides more detailed information compared to bounding boxes. There are three main image segmentation tasks: semantic segmentation groups similar objects into classes, instance segmentation classifies pixels based on instances or overlapping objects, and panoptic segmentation combines semantic and instance segmentation for comprehensive image analysis and tracking of objects. These tasks enhance computer vision capabilities by providing accurate information about object location, size, and number [36, 40].

2.1.3 YOLO - Object Detection Model

The YOLO (You Only Look Once) object detection model is a popular and influential approach in computer vision. Unlike traditional object detection algorithms that require multiple passes over an image, YOLO performs object detection in a single pass [41]. It divides the input image into a grid and predicts bounding boxes and class probabilities for objects within each grid cell. YOLO utilizes a deep convolutional neural network to predict object classes and refine bounding box coordinates simultaneously [42]. It achieves real-time object detection by optimizing the trade-off between accuracy and speed [43]. YOLO has gained significant attention for its ability to detect objects in real-time videos and images, making it highly applicable in various applications such as autonomous driving, surveillance systems, and augmented reality [41].

2.2 Photography terminology

Photography encompasses various technical aspects, including composition, capture, and post-production. This section provides a concise overview of the photographer's workflow, covering the process from initial composition to final editing while also clarifying technical terminology commonly employed in the field of photography.

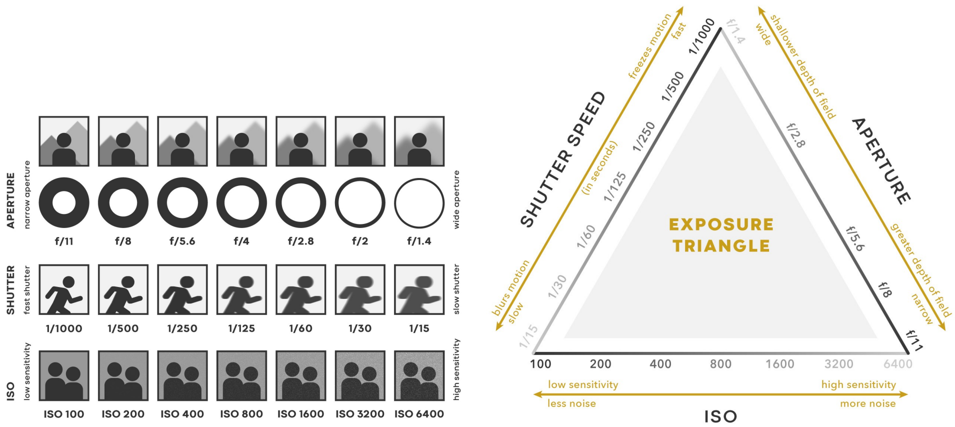
2.2.1 The Exposure Triangle

Three essential components must be considered to compose a photograph effectively: ISO, aperture, and shutter speed. These components, collectively known as the exposure triangle, are interrelated and require careful adjustment to achieve optimal results. This is visually presented in Figure 2.3b [44].

ISO refers to the camera's sensitivity to light and is crucial for converting incoming light into a digital photograph [45]. It ranges from low to high sensitivity, with low ISO preferred in well-lit settings and high ISO used in low-light environments. However, higher ISO levels introduce noise, which is visual distortion in the image and can be seen in Figure 2.3a [44].

Shutter speed determines the duration for which the shutter in the lens remains open, affecting the exposure and capturing motion. Faster shutter speeds freeze action, while slower speeds create a blurred effect. Longer shutter speeds also allow more light to reach the sensor, reducing the need for high ISO settings, which is visualized in Figure 2.3a [44].

The aperture controls the width of the lens opening, influencing the amount of light reaching the sensor. A higher aperture value represents a narrower opening, resulting in a greater depth of field and less light. Conversely, a lower aperture value means a wider opening, providing a shallow depth of field and more light. This is visually presented in Figure 2.3a [44].



(a) Exposure Chart **(b)** Exposure Triangle

Figure 2.3: How ISO, shutter speed, and aperture work and affect each other [44].

Properly tuning these three components, considering the available light and desired effects, is crucial for achieving a well-composed photograph. Incorrect settings can lead to overexposed (excessively bright) or underexposed (too dark) images, potentially causing the loss of essential details that cannot be recovered in post-production [46].

2.2.2 Image File Formats

Two primary formats are commonly used when capturing digital photographs: *RAW* and *JPEG*. Professional photographers favor the RAW format as it retains a significant amount of unprocessed data directly from the camera's sensor [47]. This format allows for extensive post-production adjustments and helps correct issues like overexposure [48]. On the other hand, the JPEG format employs lossy compression, resulting in smaller file sizes by permanently discarding data [49, 50]. It is widely used for everyday photography and immediate sharing without requiring extensive editing. While photographers often shoot in RAW, the final edited images are usually saved in JPEG, especially for web use, due to their smaller file size and broader compatibility [51].

2.2.3 Post-production

Production in photography refers to the image capture phase, while post-production encompasses all activities after the shoot until the final product [52]. A common part of post-production is a process called photo culling, where the best images are selected, and undesired ones are removed [53], and is usually done before editing to streamline the workflow [54]. Factors such as exposure, sharpness, lighting, and subject placement are considered during culling [54]. The selected photos are then edited to enhance color and exposure, aiming for the desired outcome without altering the original image. Editing is commonly performed before delivering or sharing photographs, enhancing the image while avoiding photo manipulation [55].

2.3 Related Work

This thesis explores the potential of deep learning in distinguishing between good and bad photographs, which can be viewed as automating photo culling. This section explores several AI-based systems available on the market that attempt to accomplish this task. However, none are based on research papers, making it challenging to discuss their AI aspects. Therefore, this section focuses primarily on the systems' features and how they can inspire this study.

AfterShoot [56] is a software designed to automate photo culling while giving users complete control over the selection process. Users can customize preferences for the AI's culling behavior, such as the strictness in removing blurred photos and grouping duplicates based on similarity. They can also specify the number of duplicates to be presented, such as a percentage or the top-rated ones. The AI in AfterShoot is trained to identify objective issues (e.g., blurriness) and subjective aspects like lighting, aesthetics, and composition. It clusters similar images and assigns scores to each group before selecting the best based on user-defined parameters [57]. At least one photo from each duplicate set is chosen to represent all moments [58]. The results are displayed as group duplicates with a scoring system of one to five stars, allowing users to view and select images individually or collectively based on the ratings. AfterShoot offers additional features like "Sneak Previews," which compares selected images against top-performing social media posts, and "Key Faces," which zooms in on faces to facilitate easy evaluation [57, 56]. While the program works best with human images, it has also shown promising performance with product and landscape photos [59].

Narrative Select [60] is an AI-assisted software for photo culling that allows users to organize their photos into "scenes" or similar image groups [61]. Instead of fully automating the process, users can manually rate their photos using a five-star and color-based rating system [62]. The software employs AI for face detection and assessment, enabling zoomed-in views of faces, focus and eye evaluations, and expression analysis. These assessments are presented through symbols and color grading, allowing users to evaluate and cull images quickly [62]. Additionally, Narrative Select uses AI for image assessment, indicating the "worst" images within a scene with yellow or red hexagons, helping users focus on selecting the best images [60, 62]. The aim is to narrow the selection to 20-30% of the images, streamlining the culling process [60].

Photo Culling [63] is a mobile app developed by Canon U.S.A., Inc. It utilizes Canon's Computer Vision Artificial Intelligence Engine, called "PHIL", to assess and score the user's photos based on factors like sharpness, noise, emotions, and closed eyes. The app suggests deleting duplicate photos to save storage space on the user's phone. Two culling methods are available: whole culling and similar culling. In whole culling, the AI selects photos with scores above the user-set threshold [63].

In contrast, similar to culling, it identifies the best images within a group and recommends deleting the rest. The app also offers event albums and a photo count on the home screen [63]. It appears Photo Culling aims to assist everyday photographers in managing their photo collection and freeing up phone space.

Adobe Lightroom [64] is a widely used photo editing software that has become an industry-standard [65]. While it was not initially designed for photo culling, many users utilize its built-in rating systems [66]. Adobe Lightroom does not have specific AI features for the culling process, as its focus is primarily on photo editing. Users can use the flag system to mark images for editing or rejection and apply a five-star rating system for further categorization. The advantage of using Adobe Lightroom for culling is the seamless transition to editing the selected images after applying the ratings [66].

This section explores and analyzes existing AI-based systems for automating photo culling. It discusses their features and approaches, offering insights and inspiration for developing a photography rating system. By examining the capabilities and functionalities of these systems, valuable lessons can be learned regarding the design and implementation. The section highlights the importance of customizable AI models, manual rating systems with AI assistance, image assessment techniques, and seamless integration with editing tools. This knowledge can inform the development of a similar system to ensure it incorporates practical features and addresses the needs and preferences of users.

2.4 Summary

This chapter delves into deep learning and imaging techniques for analysis, the various components that contribute to composing a photograph, from camera settings to post-production, and an exploration of existing AI-based systems in the market designed for photo culling. These sections provide essential background knowledge and serve as a driving force for designing a system that utilizes deep learning techniques to rate images. By examining deep learning and its application in image analysis, insight is gained into the technical aspects of defining a good photograph. Additionally, studying existing systems allows for understanding their strengths and limitations, ultimately inspiring the integration of these two domains—deep learning and photography—for a comprehensive understanding and motivation in this thesis.

3 Methodology

The research undertaken in this thesis adopts a *qualitative research approach*. This chapter aims to provide a comprehensive overview of the available methodology methods for qualitative research, explaining the selected approach and the rationale behind its adoption. The methodology section will adhere to the structure outlined in "The Portal of Research Methods and Methodologies" developed by Anne Håkansson [26], which serves as a guiding framework for organizing and presenting the research methods employed in this study.

3.1 Philosophical Assumptions

The choice of a philosophical assumption is crucial as it shapes the entire research process, including the selection of appropriate methods. There are various paradigms, such as positivism, realism, interpretivism, and criticalism. *Interpretivism* emphasizes that reality is constructed through social interactions and focuses on exploring the depth and complexity of phenomena to understand the meanings assigned by individuals. This approach, often used in projects involving opinions, perspectives, and experiences, is particularly effective in the development of computer systems and artifacts [26]. This aligned with the study being performed in this thesis, as the goal is to develop a computer system.

3.2 Research Methods

Research methods are crucial in supporting various stages of a research task. For qualitative research, six common methods are utilized: fundamental, applied, non-experimental, empirical, analytical, and conceptual. For this thesis, three research methods were considered: empirical, analytical, and conceptual. The *empirical research* method utilizes observations and experiences to gather data, analyze relationships, and generate knowledge, employing both quantitative and qualitative methods to explain the observed situations. *Analytical research* method involves testing pre-planned hypotheses using existing knowledge and collected data to critically evaluate and make informed decisions, particularly in areas like product design and process design. *Conceptual research* method focuses on developing new concepts or interpreting existing concepts through theory development, historical research, literature reviews, and critical analysis. It aids in establishing concepts in a specific area and involves studying literature to analyze and interpret commonly-used concepts rather than simply providing background data [26].

While the ultimate objective of this project is to build a deep learning-based system for determining whether an image is good, the research approach primarily aligns with conceptual research rather than analytical or empirical methods. This is because the study focuses on exploring concepts within deep learning, leveraging existing knowledge to conceptualize the system's development. Thus, the research method employed is *conceptual research*.

3.3 Research Approaches

Research approaches, namely inductive and deductive, are commonly employed to conclude qualitative and quantitative research. Inductive reasoning, associated with qualitative methods, focuses on developing theories by comprehending and establishing various perspectives of a phenomenon, including the possibility of developing artifacts. On the other hand, deductive reasoning, linked to quantitative methods, aims to test theories by confirming or refuting them [26].

In line with the qualitative nature of this project, an *inductive approach* will be utilized to develop a theory through understanding existing phenomena in deep learning.

3.4 Research Strategy/Design

Research strategies and designs serve as the methodologies that provide guidelines for organizing, planning, designing, and conducting research [26]. The *exploratory research* method aims to discover relationships between variables and generate general findings by utilizing surveys and qualitative data collection to gain insights into the research question. However, it does not typically provide definitive answers to specific issues. The *ethnography method*, derived from anthropology, involves studying people and cultures through descriptive research. It focuses on understanding the social and cultural context of phenomena under investigation, aiming to provide a comprehensive portrait of the subjects [26].

Exploratory research was chosen for this thesis to delve into the topic and uncover new insights. It provided flexibility to explore different aspects, generate ideas for further research, and develop new approaches.

3.5 Data Collection Methods

Data collection methods are employed to gather information to conduct research. *Language and text analysis* are utilized to interpret and analyze discourse and conversations and extract meanings from various texts and documents [26]. Language and text data allow researchers to explore diverse perspectives, experiences, and contexts in an open-ended fashion, enabling a comprehensive investigation [67].

More specifically, within language and text analysis, there is *document analysis*. Existing textual materials like documents, reports, articles, or archival sources are carefully examined. By extracting pertinent information and insights from these texts, researchers can develop an initial comprehension of the subject matter and identify potential directions for further investigation [67]. *Document analysis* was chosen for this study, and the data collected for this study consists of information related to deep learning technology, predominantly obtained through literature research and online investigation of existing systems. This data, acquired through these methods, aims to address the research question and provide insights into constructing a system capable of determining good photographs.

3.6 Data Analysis Methods

Data analysis methods are employed to examine and process the collected material, involving inspection, cleansing, transformation, and data modeling. These methods facilitate decision-making and draw meaningful conclusions from the data [26]. One of the data analysis methods is *coding*, which involves naming and labeling concepts and strategies identified during data collection to transform qualitative data into quantitative data. *Narrative Analysis* involves the examination and interpretation of literary texts and can also be utilized to facilitate traceability in requirements and interfaces [26].

In this research, *coding* is used to identify and label the concepts and strategies discussed in the related work section, aiming to understand the techniques employed by existing systems in the market. Although statistical analysis may not be applied, the focus is on identifying the techniques used by these systems. Building upon the mapping of existing systems done utilizing coding, *narrative analysis* is utilized to gain insight into the needs and motivations used to design a system. Resulting in a deeper understanding of user perspectives to create informed design decisions,

3.7 Quality Assurance

Validating and verifying the research material is crucial for ensuring quality assurance in qualitative research. The data collected for this thesis on deep learning is obtained from reliable web sources and textbooks while acknowledging that peer-reviewed articles are not available to support the data collected from the analysis of existing systems.

Transparency is maintained throughout the research process, with all relevant information being presented and no withholding of information for personal gains. This ensures that decisions made in this thesis are based on accounted-for inspirations.

3.8 Dataset

A well-curated dataset is of paramount importance in a deep learning project. The quality and composition of the dataset directly impact the performance and generalization ability of the deep learning model. A carefully curated dataset ensures that the model is exposed to diverse and representative samples, capturing the variability and complexity of real-world data. Including a wide range of examples enables the model to predict unseen data accurately. Additionally, a well-curated dataset helps mitigate biases and ensure fairness in the model's outcomes by considering different demographic factors and avoiding underrepresenting or overrepresenting specific groups. A well-curated dataset lays a solid foundation for training deep learning models, fostering improved performance, generalization, fairness, and reliability in their predictions.

A photographer with eight-year field experience has assembled the dataset curated for this thesis. With a deep understanding of the nuances and requirements of photography, the curator hand-picked each image to ensure a wide variety of representations. The dataset encompasses diverse subjects, lighting conditions, compositions, and styles, capturing the rich complexity of real-world photographic scenarios. By drawing from a broad range of images, the curated dataset provides the model with ample training examples to learn robust and discriminative features. This comprehensive curation process helps to ensure that the model is exposed to a diverse set of images, enabling it to generalize well and make accurate predictions on a wide range of photographs.

The dataset employed in this study comprises 2,370 images, which have been further organized into seven distinct sub-datasets, each dedicated to training a specific model. The subsequent chapters of the thesis will provide a comprehensive explanation regarding the precise division and categorization of the dataset, offering more profound insights into this aspect of the research.

3.9 Summary of Development Tools

DeepRoom's implementation utilized various development tools highlighted in Table 3.1. This table provides an overview of the specific tools employed during the development process of DeepRoom.

Anaconda [68] was employed as a distribution platform to set up the coding environment, ensuring consistent and easy setup across the team. Jupyter [69] served as the computing platform. It enables developers to write and execute code interactively, providing a convenient environment for experimentation and iteration [69]. However, for this project, it was just used as a hosting platform for the implementation. TensorFlow [70], the core deep learning framework, formed the foundation for building and training convolutional neural network models. Keras [71], built on top of TensorFlow, offered a high-level API that simplified the construction of convolutional neural networks (CNNs) for the project. Matplotlib's pyplot [72] was utilized as a visualization tool, allowing the plotting of evaluation graphs to analyze the performance of the models. Seaborn [73], another visualization tool, was explicitly employed for generating classification reports, providing insights into the accuracy and performance of the classification tasks. Lastly, Roboflow [74] played a crucial role in the development process by providing software for labeling and annotations and training the object detection model, streamlining the computer vision model training pipeline.

Table 3.1: Implementation development tools.

Software	**Area of use**
Anaconda	Distribution platform for setting up the coding environment.
Jupyter	The computing platform used to write and execute code.
TensorFlow	The deep learning framework.
Keras	The library, built on top of TensorFlow, was used to construct the CNNs.
Matplotlib's pyplot	Visualization tool used to plot evaluation graphs.
Seaborn	Visualization tool used for classification report.
Roboflow	Software used for labeling and annotations, and training of object detection model

Combining these tools and software in the development process of DeepRoom ensured a comprehensive and efficient approach to implementing and evaluating the deep learning models for the project.

4

Design

This chapter will describe the design of *DeepRoom*, a deep learning rating system for photography, based upon the findings from Chapter 2. It will detail the requirements of building a system as proposed in Section 1.2, a DL-based system to determine good photographs. It will explore the key factors contributing to a good or relevant photograph based on the technical aspects behind the composure of a photograph, as discussed in Section 2.2. Further on, it will describe the underlying architecture of the system and how DL techniques, as discussed in Section 2.1, can be utilized to fulfill the requirements. Lastly, it will propose two approaches to deploying the DeepRoom for real-life use using the same underlying architecture.

This chapter is based on the design proposed in the capstone project "Determining the Relevance of Photographs: A Deep Learning Approach" written by Victoria Kumetz Lillegård Lintvedt in 2022 that this thesis implements [27].

4.1 Requirements

The requirements presented in this section are based on the problem presented in Section 1.2. This section describes the DeepRoom system's minimum requirements, expected functionality, and additional features to improve further and optimize the system. Each requirement is discussed in detail to provide clarity on its meaning and purpose:

Minimal Requirements:

- **Utilization of powerful image recognition technique for photo analysis:** The project's core objective is to employ deep learning (DL) methods to determine the quality of photos. Thus, the system must utilize DL algorithms to analyze the photographs based on the factors, which will be discussed further in the next section, to assess their relevance.

- **Assignment of ratings to each photo:** The system should assign a rating ranging from one to ten to each analyzed photo. This rating system enables the identification and presentation of the best photographs within the dataset to the end user.

- **Differentiate between animate and inanimate photos:** Certain factors vary for animate and inanimate photography. Hence, the system needs to handle these distinctions appropriately. Furthermore, the system should be able to handle photographs that feature subjects not being the image's primary focus.

- **File format compatibility:** Considering the popularity of RAW and JPEG file formats in photography (as discussed in Section 2.2), the system should, at a minimum, support these file formats to accommodate a wide range of photo files.

Additional features:

- **Handling large datasets:** The system should be designed to handle and process large datasets effectively, ensuring efficient performance and scalability.

- **Detection and grouping of duplicates:** The system should possess the capability to detect and group similar photos together, even if they are not exact duplicates. This feature facilitates the analysis and rating process by directly comparing similar photos.

- **Personalized culling preferences:** The system should allow end users to personalize their photo culling process. This may include preferences such as displaying only the worst photos (those recommended for deletion) or only the best photos (those recommended for further post-production).

The DeepRoom system must prioritize the essential functionalities of analyzing and rating photos, differentiating between animate and inanimate photos, and supporting various file formats. These core features are crucial for its effective operation. Additionally, the system can further enhance its capabilities by handling duplicates, managing large datasets, and offering personalization options to cater to the specific needs and preferences of the end user. The core functionalities of analyzing and rating photos and differentiating between animate and inanimate photos are implemented as the fundamental components of the DeepRoom system. These essential features are the backbone of the system's functionality and are prioritized in the implementation process, while the additional functionalities are considered supplementary enhancements.

4.2 Photograph Rating

A scale of one to ten will establish a rating system for the photographs. The evaluation of factors for the rating system will be approached from a technical standpoint, considering the discussions in Chapter 2.2. The evaluation will vary based on whether the photograph contains animate subjects (e.g., humans or animals) or inanimate subjects (e.g., landscapes or cityscapes). The following sections present and explain the different factors considered for the rating system:

- **Exposure:** If an image is excessively over- or underexposed to the extent that no data can be recovered from the incorrectly exposed areas, it could be considered irrelevant. Such a situation may occur due to improper usage of the exposure triangle while capturing the image.

- **Focus & sharpness:** The focus of an image is related to the physical focusing of the camera lens, whether it is manually or automatically controlled. An image that is out of focus appears blurry and lacks sharpness. Incorrect aperture or shutter speed settings during photography can also contribute to a lack of sharpness.

  Analyzing the focus is a crucial factor in determining the relevance of a photograph, and different criteria apply to animate and inanimate photos:

  - **Animate**: The subject is expected to be focused and sharp for images that include a subject, such as a person or an animal. In the case of multiple subjects, the majority of the subjects should be in focus and sharp. When the subject is positioned prominently in the center of the image, as is often the case in portrait photography, the photographer typically intends for the subject to be the image's primary focus.

- **Inanimate**: If an image does not have a specific subject to focus on, the analysis should shift toward different image segments, such as the foreground, middle ground, or background. It is important to ensure that at least one of these segments is in focus and sharp. These segments represent areas within the image where focus and sharpness are desired. Alternatively, this criterion should also be considered if the intention is to have the entire picture in focus and sharp.

- **Horizon lines:** This factor primarily applies to inanimate photos, particularly landscape photography, where a visible horizon or straight lines within the photo (e.g., buildings) can appear skewed. The straightness of a photograph can always be adjusted during post-production. However, when the image is straightened, certain edges may be lost, which can be undesirable if there are essential elements near the edges that the photographer intends to include in the final composition.

Table 4.1: Photo rating values. Collected from the capstone project "Determining the Relevance of Photographs: A Deep Learning Approach" written by Lintvedt, Victoria K. L. in 2022.

Rating	**1-2**	**3-4**	**5-6**	**7-8**	**9-10**
Overexposure	>30%	20-30%	10-20%	5-10%	0-5%
Underexposure	>30%	20-30%	10-20%	5-10%	0-5%
Focus & Sharpness (animate)	<50%	50-70%	70-85%	85-95%	95-100%
Focus & Sharpness (inanimate)	<40%	40-60%	60-75%	75-90%	90-100%
Horizon Lines	> 15°	10-15°	5-10°	1-5°	0-1°

Table 4.1 presents a preliminary overview of how photos could be rated from a technical standpoint. Ratings ranging from one to four will be considered as "below average," five to six as "average," and seven to ten as "above average." Each factor will be assessed separately for each photo, distinguishing between animate and inanimate categories. An overall rating will then be determined based on the average scores of the individual factors.

Focus and sharpness are interrelated factors that differ between animate and inanimate images. For animate subjects, the evaluation focuses on the subject(s). The rating will be average if the entire subject is focused, higher if the subject's face is in focus, and best if the eyes are focused. Sharpness is assessed based on the focused area, and the percentage of sharpness determines the rating. In the case of inanimate subjects, three elements are considered: foreground, middle ground, and background. The highest rating is given when all three elements are in focus, the next best rating when two are in focus, and an average rating when only one element is in focus. Similar to animate photos, sharpness is correlated with the focused areas, and the rating is based on the percentage of sharpness. Any photo with below-average focus will receive a low rating. Horizon lines are evaluated based on how many degrees the horizon deviates from being level, and the rating is determined accordingly.

Each factor is individually assessed before calculating the average scores, determining the final rating assigned to an image. Images falling into the below-average category are recommended for deletion by the system. Images rated as average require further review, while those receiving an above-average rating are considered relevant. This establishes a cutoff for images rated below average, but this only applies to the overall rating, not the individual factors. For instance, an image that scores above average on all factors except horizon lines will still receive an above-average rating overall. These ratings aid in determining the relevance of a photograph.

These factors serve as guidelines for assessing the relevance of a photograph. Some factors may vary between animate and inanimate photographs due to differences in focal points. It is also essential to consider that some photographs may include subjects without them being the main focus of the composition, and this aspect should be addressed in the system's design.

4.3 Architecture

The heart of the DeepRoom system lies in its profound functionality embodied by deep learning models. These models serve as the fundamental tools responsible for rating and evaluating images. As explained in the preceding sections, the DeepRoom system employs a nuanced differentiation between animate and inanimate subjects, presenting the first model—an object detection model capable of identifying and categorizing animate subjects. Consequently, this distinction leads to categorizing images into two categories: those featuring animate subjects and those without.

Following this classification, dedicated deep learning models are employed to analyze animate and inanimate images, respectively. The inanimate models undergo training to detect various quality aspects such as blurriness, skewed horizons, overexposure, and underexposure. By considering the entirety of the photograph, distinct models are developed for each quality factor. On the other hand, the animate models, sharing similarities with their inanimate counterparts but excluding the consideration of skewed horizons, are honed to concentrate on these quality aspects, specifically within the subject itself.

This intricate approach highlights the system's attention to detail and the multi-faceted nature of the deep learning models employed in the DeepRoom system. By incorporating separate models for animate and inanimate images and training them to focus on specific aspects, the system demonstrates a comprehensive understanding of image evaluation, allowing for precise analysis and accurate rating. The architecture is presented in Figure 4.1.

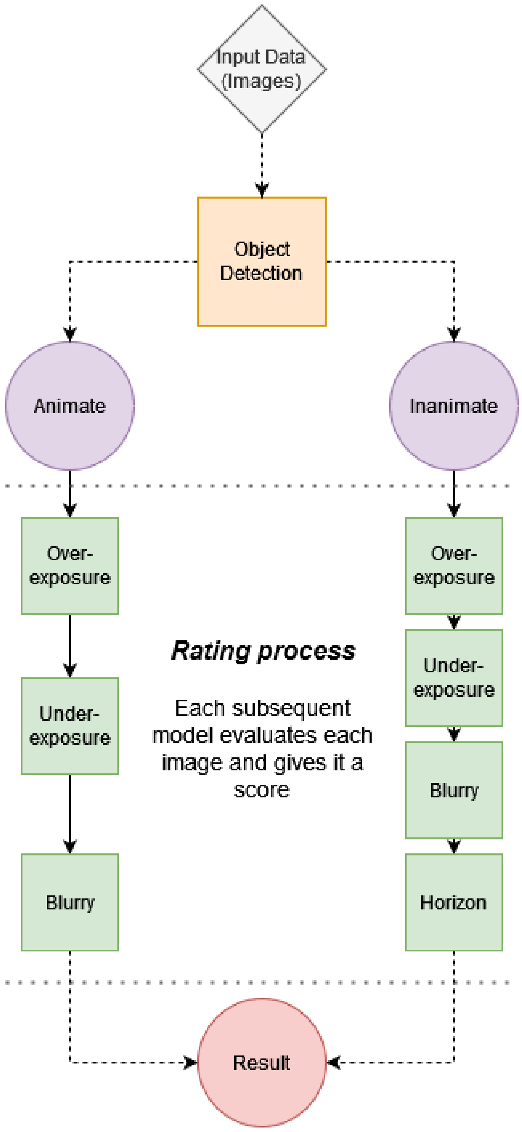


Figure 4.1: DeepRoom's architectural overview.

4.4 Deployment

This section will explore a console-based deployment option specifically designed for servers or non-graphical operating systems, enabling efficient utilization in a command-line interface. This streamlined approach simplifies the system's deployment process.

When using DeepRoom, users can select a folder containing the images they wish to rate. The system applies the underlying deep learning architecture discussed in Section 3.2 to rate all the images within the chosen folder. DeepRoom incorporates a predetermined "cutoff" rating, such as a score of 6, as a threshold. Images scoring equal to or above the cutoff are moved to a folder labeled "keep," while those below the cutoff are moved to a folder named "delete." It is important to note that these suggestions are intended as guidance, allowing users to review and decide on the proposed "bad" images.

Furthermore, the console-based system of DeepRoom includes customizable settings that provide flexibility to users. For instance, users can overwrite the rating method for all images, opting for either the animate or inanimate rating. The cutoff limit can also be adjusted according to specific preferences and requirements. These settings empower users to tailor the rating process to their needs and preferences.

5

Implementation

This chapter provides an overview of the implementation of DeepRoom, focusing on key details and processes. It discusses the development tools, provides further insight into the dataset used, and the steps taken for data preprocessing. Additionally, it explores the construction and training of each deep learning model employed in the project. By examining these implementation-specific aspects, a deeper understanding of DeepRoom's inner workings is attained.

5.1 Development Tools

The implementation of the project required a combination of software tools and hardware resources to support the development and training of deep learning models. These tools and hardware were essential in various project stages, including data preprocessing, model design, training, and evaluation. The software tools provided the necessary frameworks and libraries to build and deploy deep learning models, and the hardware resources ensured sufficient computational power to handle complex computations.

5.1.1 Software

During the implementation of the project, several development tools were utilized to facilitate the creation and training of deep learning models. These tools played a crucial role in various stages of the development process, including data preprocessing, model architecture design, training, and evaluation.

One of the primary tools employed was *Anaconda* [68], a popular open-source distribution platform that simplifies package management and environment setup [75]. Anaconda provided a seamless way to manage different Python libraries and dependencies required for the project, ensuring compatibility and reproducibility across different systems.

TensorFlow [70] and *Keras* [71] were employed as the core deep learning frameworks. TensorFlow, an open-source library, offered a comprehensive set of tools and functionalities for building and training deep learning models. Keras, built on top of TensorFlow, provided an easy way of constructing neural networks, simplifying the implementation process and allowing for faster prototyping.

Jupyter [69] notebooks served as an essential tool for coding and experimentation. It provided an efficient environment for developing and testing code snippets, allowing for easy data exploration and visualization.

For data visualization, *Matplotlib's Pyplot* [72], and *Seaborn* [73] were utilized. It provided an easy way to analyze and present the results of the models and aided in gaining insights into the performance and behavior of the deep learning models.

Lastly, *Roboflow* [74] played a vital role in annotating and training the object detection model. Roboflow offered a user-friendly interface for annotating and labeling images, making it easier to generate training datasets. It also provided data augmentation and preprocessing functionality, enhancing the diversity and quality of the training data.

Together, these development tools formed a robust ecosystem that facilitated the implementation of the project. They streamlined the development process, empowered efficient experimentation, and enabled the creation of high-performance deep learning models for the task at hand.

5.1.2 Hardware

The deep learning models in this project were trained using the following hardware specifications: an *AMD Ryzen 7 3700X 8-Core Processor* [76] as the CPU and an *Nvidia GeForce GTX 1070* [77] as the GPU. With its multi-core architecture and powerful processing capabilities, the CPU provided efficient execution of computations during training. The GPU, known for its parallel processing capabilities, significantly accelerated matrix calculations and neural network operations, enhancing the training process. The *32GB of RAM* allowed for support of the training process and handling large datasets and model parameters, ensuring smooth and efficient data loading.

5.2 Object Detection Model

The development of the Object Detection model involved utilizing the YOLOv8 Object Detection model architecture. The computer vision platform Roboflow was employed to annotate and train the model, offering helpful tools for easy and precise annotation and efficient training. The model's objective is to detect two distinct categories: humans and animals. For this purpose, a dataset of 270 images encompassing human and animal instances was prepared and employed as the foundation for training. The dataset consisted of 256 instances featuring humans and 137 instances featuring animals.

The dataset was divided into three sets: the training set, encompassing 68% of the data, served as the primary component during the model's training phase. The validation set, comprising 16% of the data, played a crucial role in the iterative refinement of the model. The model's weights and biases were adjusted by leveraging the validation set's results, enabling enhancement and optimization of the model. Lastly, the testing set, accounting for 16% of the data, acted as the final evaluation benchmark for the trained model.

Table 5.1 provides an overview of the dataset setup and includes specific details.

Table 5.1: Object detection dataset details.

No. Images	Labels	Class Balance	Train/Validation/ Test Split
270	human & animal	256 with humans, 137 with animals	68% / 16% / 16%

5.3 The CNN Rating Models

In total, there are six different models built for rating the photographs. The rating models, as shown in Figure 4.1, were developed from scratch using Keras to construct Convolutional Neural Networks (CNNs) for extracting image features. Six distinct CNN models were trained to detect specific attributes and rate the images accordingly. These attributes included Blurry, Overexposed Animate, Overexposed Inanimate, Underexposed Animate, Underexposed Inanimate, and Skewed Horizon for inanimate images. While the overall structure of the CNN models remained similar, slight adjustments were made to cater to the unique characteristics of each attribute.

5.3.1 Preprocessing

All the rating models adhere to a consistent structure of data preprocessing and can be seen in Table 5.2. Initially, all images are resized to a uniform dimension of 224x224 pixels and undergo rescaling to ensure normalization. To prevent memory overflow, a batch size of 32 is established. An image generator is also employed to introduce variations and augment the dataset, enhancing its robustness. Each model's dataset is further divided, allocating 90% for training purposes and reserving 10% for validation. This partitioning facilitates practical model training and evaluation.

Table 5.2: Dataset preprocessing details

Train/Validation Split	Batch Size	Resize Shape
90% Train & 10% Validation	32	224x224 pixels

5.3.2 The Over- and Underexposure Models

The architectural design of the rating models encompasses four distinct CNNs, tailored explicitly for analyzing over- and underexposure in both animate and inanimate photographs. While the fundamental structure remains consistent, the differentiation lies in the datasets curated for each model. The inanimate model's datasets comprise images primarily showing significant over- or underexposure, while the animate model's datasets focus explicitly on subjects being over- or underexposed. Each dataset consists of approximately 300 images, equally divided into "good" and over-/underexposed categories, serving as the training labels for the models. Table 5.3 provides an overview of the dataset setup and includes specific details.

Table 5.3: Exposure datasets details.

	No. Images	Labels (and value)
Overexposure Animate	300	good (1) & overexposed (0)
Overexposure Inanimate	300	good (1) & overexposed (0)
Underexposure Animate	300	good (1) & underexposed (0)
Underexposure Inanimate	300	good (1) & underexposed (0)

The model construction, depicted in Figure 5.1, entails three convolutional layers followed by flattening, leading to two dense layers. A dropout layer is incorporated to randomly deactivate 10% of the neurons during training, mitigating the risk of overfitting.

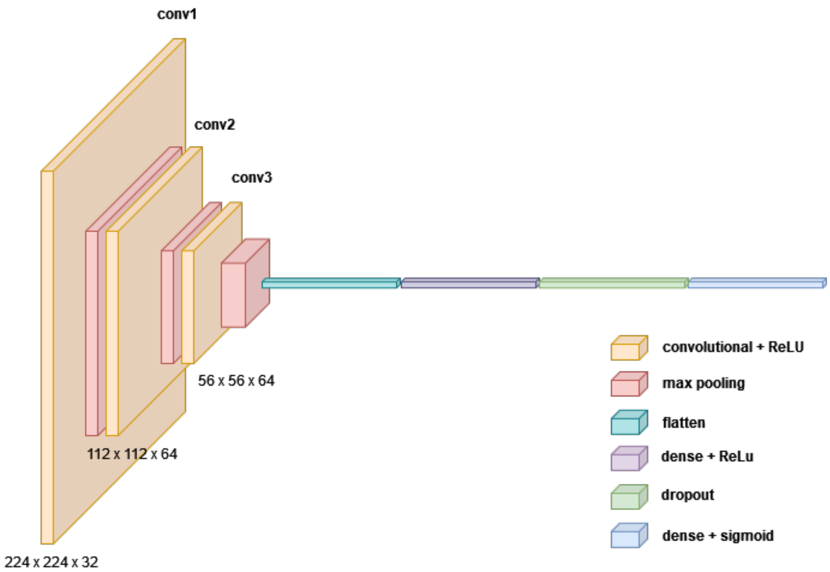


Figure 5.1: CNN model for exposure detection

5.3.3 The Blurry Model & The Skewed Horizon Model

The blurry model is trained on a dataset comprising 600 images, encompassing both animate and inanimate photographs. Half of these images are classified as "good," while the remaining half represents blurry photographs, which serve as the labels for the model's training. The skewed horizon model is specifically designed for inanimate images. Its dataset comprises 300 images, similar to the distribution in the blurry model, with half of the images labeled as "good" and the other half labeled as "skewed horizon." Table 5.4 provides an overview of the dataset setup and includes specific details.

Table 5.4: Blurry and skewed horizon datasets details.

	No. Images	Labels (and value)
Blurry	600	good (1) & blurry (0)
Skewed Horizon	300	good (1) & skewed-horizon (0)

Both models follow the same architecture, consisting of two convolutional layers, flattening, and two dense layers. The model is depicted in Figure 5.2. In contrast to the exposure CNN architecture, the CNN architecture utilized in this case does not incorporate a dropout layer. The concern for overfitting was comparatively less pronounced, leading to the exclusion of this layer. Additionally, the architecture does not feature a max pooling operation following the final convolutional layer. These two distinctions, the absence of dropout and max pooling, are the only factors that differentiate the architecture depicted in Figure 5.1 from the one in Figure 5.2. Apart from these variances, both models' architectures are identical in their structure and employ the same activation functions.

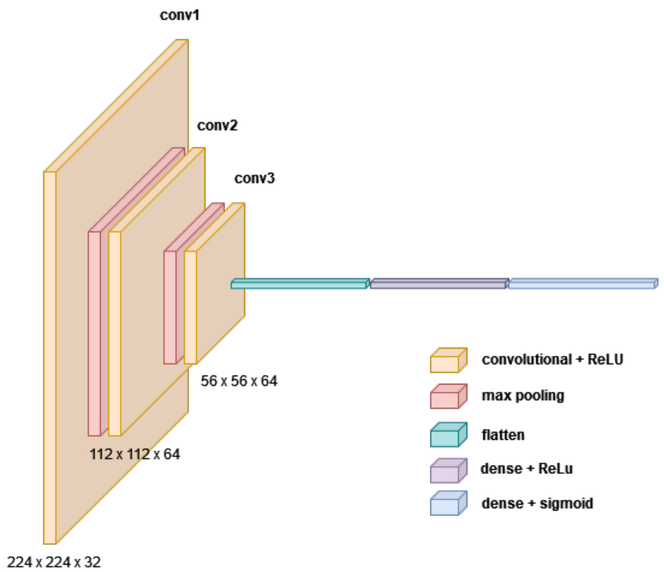


Figure 5.2: CNN model for blurry and skewed horizon detection

5.3.4 Rating Evaluation

The model generates a confidence score (C) ranging from zero to one for each photograph, representing its prediction. This score is the basis for assigning a rating (R) to the photograph, calculated by multiplying the confidence score by ten. The rating calculation is illustrated in Equation 5.1. For instance, a prediction score of 0.98 would correspond to a rating of 9.8.

$$R = C \cdot 10 \tag{5.1}$$

Furthermore, it is essential to note that each photo receives ratings from multiple models, each assessing specific features. These individual rating scores are then averaged to obtain the final rating score (FR). Equation 5.2 demonstrates the calculation of the final rating score for animate images, while Equation 5.3 presents the calculation for inanimate images. For example, if an image receives ratings of 9.8 for overexposure, 8.9 for underexposure, and 7 for blurriness, the overall rating would be 8.6.

$$FR = \frac{(R_{overexposure} + R_{underexposure} + R_{blurry}}{3} \tag{5.2}$$

$$FR = \frac{(R_{overexposure} + R_{underexposure} + R_{blurry} + R_{skewed-horizon})}{4} \tag{5.3}$$

6

Evaluation

This chapter provides a comprehensive evaluation of the DeepRoom system, focusing on each deep learning model's performance and effectiveness. The evaluation includes an analysis and discussion of the results obtained from each model, considering metrics such as accuracy, precision, and recall. The strengths and limitations of the models are highlighted, shedding light on their capabilities and potential areas for improvement.

In addition to the internal evaluation, DeepRoom is compared to similar software solutions discussed in Section 2.3. This comparison allows for a deeper understanding of DeepRoom's unique features, advantages, and potential contributions to the field. By examining the similarities and differences between DeepRoom and existing tools, valuable insights can be gained regarding its novelty, performance, and potential for practical applications.

6.1 Object Detection Model

Mean Average Precision (mAP) is a commonly used metric for evaluating the performance of object detection models. It measures the accuracy and precision of the model in identifying and localizing objects within an image. In object detection, *precision* refers to the percentage of correctly identified objects out of all the predicted objects. At the same time, *recall* measures the percentage of correctly identified objects out of all the ground truth objects. Average Precision

(AP) is calculated by taking the precision-recall curve and computing the area under the curve. The mAP is then obtained by averaging the AP values across different object classes. It provides a single value that summarizes the overall performance of the object detection model. A higher mAP indicates better accuracy and localization of objects by the model [78].

In Table 6.1, the training results of the object detection model are presented. The mAP of 84.9% indicates the model's overall performance in terms of object detection accuracy. A higher mAP value suggests that the model effectively identifies and localizes objects. Additionally, the model achieves a precision of 85.7% and a recall of 83.9%, indicating a relatively high level of accuracy in correctly detecting objects and minimizing false positives and false negatives. These results demonstrate the model's proficiency in recognizing and precisely locating objects within the training and validation dataset.

Table 6.1: Object detection training results.

mAP	Precision	Recall
84.9%	85.7%	83.9%

Figure 6.1 illustrates the progression of mAP values during the training process, displaying a generally linear graph. Notably, spikes indicate initial fluctuations and a period of stabilization required to establish a consistent mAP for the model. Towards the latter stages of training, the line flattens, indicating a more stable and consistent mAP result. This observation highlights the model's progression and improvement, achieving reliable and steady performance.

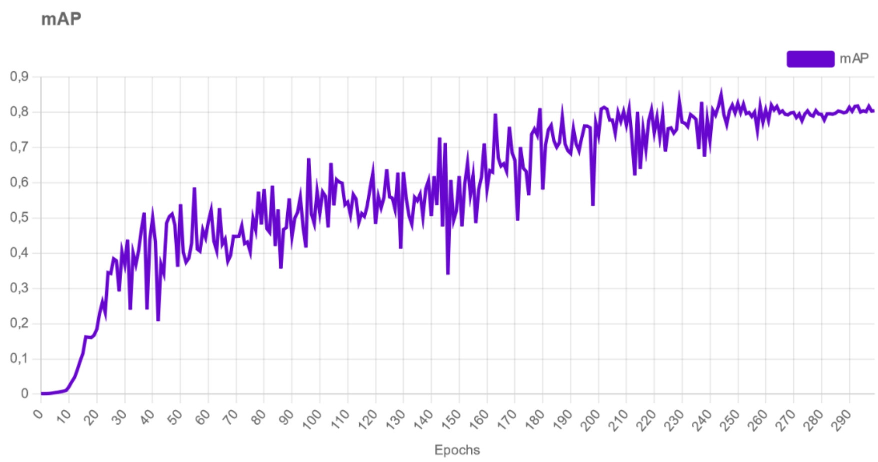


Figure 6.1: mAP graph of the object detection model.

Table 6.2 provides valuable insights into the average precision of each class in the object detection model. Notably, both the *animal* and *human* detection exhibit impressive average precision scores of 81-82%, indicating high accuracy. The exceptional performance of 96-97% in *animal* detection further emphasizes the model's capability in accurately identifying animals. However, the *human* detection score of 66-68% may be affected by annotation inconsistencies, as the bounding boxes varied from encompassing the entire human body to focusing solely on the face. In contrast, the annotation process for animals demonstrated more consistent bounding box annotations, primarily encompassing the animal's entire body.

Table 6.2: Average precision for object detection per class.

	Validation dataset	**Train dataset**
Human	66%	68%
Animal	97%	96%
All	81%	82%

In object detection, there are three main ways of measuring loss. Box loss measures the accuracy of predicted bounding box coordinates. Class loss evaluates the accuracy of object classification. Object loss determines the presence or absence of an object. These components are used to assess and improve the model's performance in detecting objects and their properties [79].

Figure 6.2 displays three graphs depicting the box, class, and object loss during training. All three graphs exhibit a downward trend, indicating a reduction in loss over time. Initially, the box loss experienced a rapid decline, followed by a more gradual decrease, ultimately reaching a relatively low score of approximately 2-3%. Similarly, the class loss demonstrates an initial steep decrease before stabilizing at less than 0.5% loss. The object loss follows a more consistent linear decrease, settling at around 1.5%. These impressive results showcase the effective learning and optimization of the object detection model.

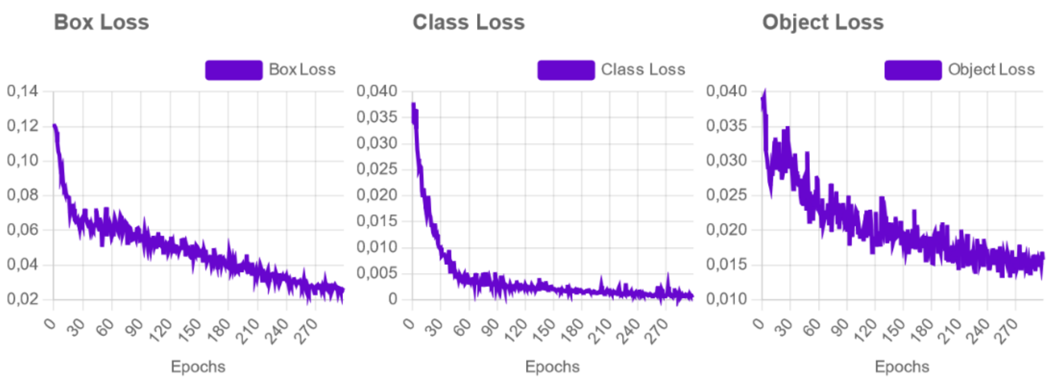


Figure 6.2: Loss graphs of the object detection model.

6.2 CNN models

When evaluating Convolutional Neural Network (CNN) models, loss and accuracy are commonly used metrics. Loss refers to the discrepancy between the predicted output of the model and the actual ground truth labels in the training data. It measures how well the model can minimize the difference between its predictions and the expected values. The goal during training is to minimize this loss, as a lower loss indicates a better fit of the model to the data. On the other hand, accuracy represents the proportion of correctly classified instances in the evaluation dataset. It measures the model's overall performance in terms of correctly predicting the class labels [80].

Apart from loss and accuracy, precision, recall, and F1-score are evaluation metrics commonly used for classification tasks. Precision measures the proportion of true positive predictions among all positive predictions [81]. It quantifies the model's ability to identify positive instances while minimizing false positives correctly [82]. Recall, also known as sensitivity or true positive rate, calculates the proportion of true positive predictions among all actual positive instances. It assesses the model's ability to capture all positive instances while minimizing false negatives. The F1-score is a harmonic mean of precision and recall, providing a single metric that combines both [81]. It provides a balanced assessment of the model's performance by simultaneously considering precision and recall [82].

By evaluating CNN models using these metrics, it is possible to assess their performance, identify strengths and weaknesses, and compare different models.

6.2.1 Training

During CNN training, the loss and accuracy values served as performance indicators. The loss values guided optimization to improve predictions by adjusting model parameters. Accuracy values reflected the proportion of correctly classified instances. Figure 6.3 illustrates the loss and accuracy of a CNN model during training, encompassing both training and validation datasets. Appendix A presents all of the training graphs of the six CNN models.

To comprehend the results, it is essential to grasp the concept of an epoch. An epoch refers to a complete iteration through the entire training dataset during the model training process [83]. In other words, an epoch represents one pass where each training sample has been presented to the model once, allowing it to make predictions and update its internal parameters based on the provided training data [84, 83].

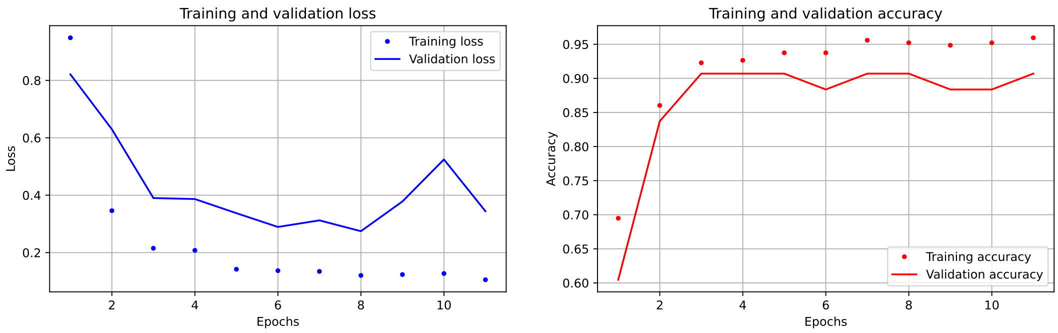


Figure 6.3: Example CNN training graphs. This is from the overexposed inanimate model.

During an epoch, the model goes through two main steps: forward propagation, where the input data is processed through the network to produce predictions, and backpropagation, where the model adjusts its internal parameters (weights and biases) based on the calculated errors between the predicted outputs and the expected outputs [34, 32, 31]. The model can learn and refine its performance by repeating this process for multiple epochs, gradually improving its ability to make accurate predictions.

Figure A.1 illustrates the training progress of the overexposed animate CNN model throughout fourteen epochs. The loss values for both the training and validation datasets remained relatively similar throughout the training process, indicating a consistent learning trend. However, a notable spike in loss occurred on the validation dataset during the twelfth epoch. In terms of accuracy, there were slight variations between the training and validation datasets, with the validation dataset generally exhibiting lower accuracy compared to the training dataset, which suggests a potential area for further investigation and improvement.

The overexposed inanimate CNN model underwent ten epochs, as depicted in Figure A.2. Throughout the training process, both the training and validation datasets exhibited fluctuating loss values. Interestingly, the validation dataset demonstrated lower loss values during the initial epochs but experienced a spike and increased loss values towards the end of training. The accuracy scores remained relatively consistent between the training and validation datasets. These findings suggest the need for further analysis to understand the causes behind the loss spike and to explore potential strategies for enhancing the model's performance.

As illustrated in Figure A.3, the underexposed animate CNN model underwent eleven epochs. The training and validation datasets exhibited remarkably similar loss and accuracy values during the training process. This consistency suggests that the model's performance remained stable and consistent across the training iterations. These findings highlight the model's ability to effectively learn and generalize patterns from the data, demonstrating its potential for accurately classifying underexposed animate photographs.

The visualization in Figure 6.7 illustrates the training process of the underexposed inanimate CNN model over eleven epochs. The training and validation datasets exhibit similar trends in loss and accuracy. However, the validation dataset demonstrates slightly inferior performance, with higher loss values and lower accuracy than the training dataset.

The blurry model underwent fifteen epochs as illustrated in Figure A.5. The training datasets exhibit a smooth descending curve in loss values while the accuracy values consistently increase. Conversely, the validation datasets experience notable spikes in loss and accuracy during training, eventually converging to values similar to those of the training dataset. In summary, the model successfully reduces loss and improves accuracy throughout training, despite intermittent fluctuations observed in the validation datasets.

The training of the skewed horizon CNN model, as depicted in Figure A.6, reveals that the loss values remain consistent without significant fluctuations after the initial epoch. Both the validation and training sets yield comparable results and exhibit a parallel trajectory during training. However, the accuracy scores demonstrate noticeable fluctuations, resulting in spikes in accuracy. Despite following a similar overall trend, the validation dataset displays more pronounced spikes than the training dataset. In conclusion, the model's training demonstrates stable loss values, while the accuracy scores exhibit variability, particularly in the validation dataset.

6.2.2 Exposure models

The evaluation results presented in Table 6.3 showcase the performance of the four exposure models based on their loss and accuracy scores using the dedicated validation datasets. The accuracy scores for all models are relatively high, ranging from 87.1% to 91.4%. These scores indicate that the models can correctly classify the exposure levels with good accuracy.

However, there is room for improvement when considering the loss values. The loss values for the models are considerably higher, with the underexposure animate model achieving the best score of 14.9%. On the other hand, the other three models exhibit loss values ranging from 33.0% to 35.9%. These elevated loss values suggest that the models' predictions deviate significantly from the true labels, indicating inconsistency and imprecision in the exposure classification.

Table 6.3: The loss and accuracy scores of each exposure CNN model. Calculated using their respective validation datasets.

	Loss	**Accuracy**
Overexposure Animate	33.0%	87.1%
Overexposure Inanimate	35.9%	90.9%
Underexposure Animate	14.9%	91.4%
Underexposure Inanimate	34.4%	90.7%

Overall, while the accuracy scores of the exposure models are promising, the relatively high loss values indicate the need for further refinement and optimization. By reducing the loss values, the models can improve their precision in predicting exposure levels and enhance their overall performance.

The following figures (Figure 6.4-Figure 6.7 depict the classification reports of each of the models, including their precision, recall, and F1-score for each of the models, calculating each score for each of their labels, accuracy, macro average, and weighted average.

In Figure 6.4, the *overexposed animate* model demonstrates precision values of 0.75 for class 0 (overexposed) and 1.00 for class 1 (good). The recall values are 1.00 for class 0 and 0.69 for class 1, suggesting that the model successfully identifies all instances of class 0 but only around 69% of class 1 instances. The F1-scores, which balance precision and recall, are 0.86 for class 0 and 0.81 for class 1. These scores indicate a good trade-off between precision and recall for both classes. The model's overall accuracy is 0.84, meaning it correctly classifies 84.0% of the instances in the evaluation set. The average macro F1-score is 0.84, reflecting an overall solid performance across classes. The weighted average F1-score, also 0.84, suggests that the model's performance is consistent across both classes, considering their respective contributions to the evaluation set.

Overall the model demonstrates reasonable performance in classifying instances into class 0 and class 1. It achieves high precision for both classes and shows good recall for class 0 but relatively lower recall for class 1. The F1-scores indicate a balanced performance, and the model's overall accuracy is 84.0%.

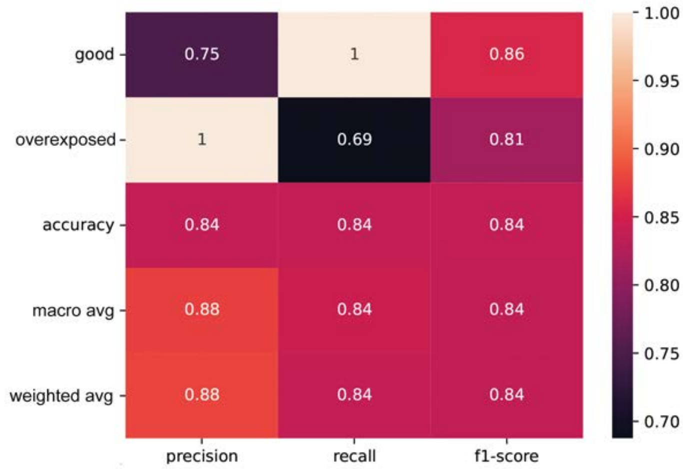


Figure 6.4: Overexposed Animate model classification report.

The *overexposed inanimate* model, depicted in Figure 6.5) demonstrates favorable performance in this report, with precision values of 0.89 for class 0 (overexposed) and 0.80 for class 1 (good). The recall values are 0.84 for class 0 and 0.86 for class 1, indicating the model's ability to identify instances of each class correctly. The F1-scores, balancing precision and recall, are 0.86 for class 0 and 0.83 for class 1. Overall, the model achieves an accuracy of 0.85, accurately classifying 85.0% of the instances in the evaluation set. The macro average and weighted average F1-scores of 0.85 suggest a consistent and robust performance across both classes. In summary, the model exhibits accurate and balanced performance, achieving high precision, recall, and F1-scores in classifying instances into the overexposed and good classes.

Figure 6.6 provides an overview of the *underexposed animate* model's performance. The model demonstrates accurate predictions for both classes with a precision of 0.86 for class 0 (underexposed) and 1.00 for class 1 (good). In terms of recall, the model achieves a recall of 1.00 for class 0, indicating it successfully identifies all instances of class 0. In contrast, for class 1, it attains a recall of 0.81, suggesting it identifies approximately 81.0% of class 1 instances. The F1-scores, which balance precision and recall, are 0.93 for class 0 and 0.90 for class 1, indicating a good trade-off between precision and recall. The model's overall accuracy is 0.91, indicating that it correctly classifies 91.0% of the instances in the evaluation set. Based on this classification report, the model successfully distinguishes between overexposed and good instances, achieving high precision, recall, and F1-scores.

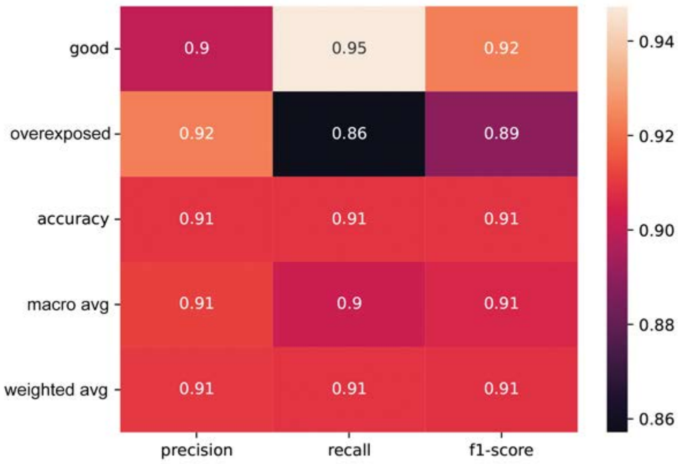


Figure 6.5: Overexposed Inanimate model classification report.

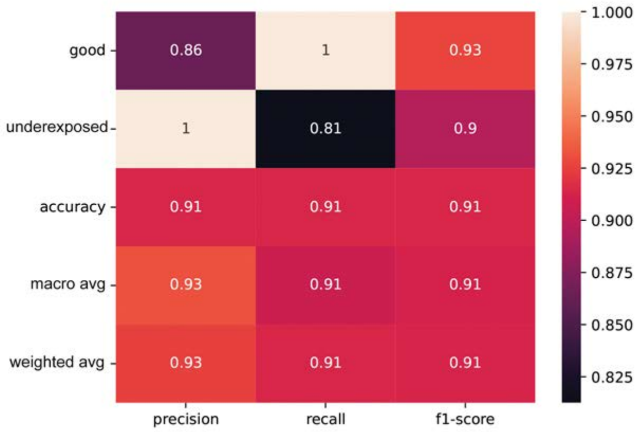


Figure 6.6: Underexposed Animate model classification report.

The *underexposed inanimate* model, depicted in Figure 6.7, demonstrates accurate predictions for both classes with a precision of 0.77 for class 0 (underexposed) and 1.00 for class 1 (good). In terms of recall, the model achieves a recall of 1.00 for class 0, indicating it successfully identifies all instances of class 0. In contrast, for class 1, it attains a recall of 0.81, suggesting it identifies approximately 81.0% of class 1 instances. The F1-scores, which balance precision and recall, are 0.87 for class 0 and 0.89 for class 1, indicating a good trade-off between precision and recall. The model's overall accuracy is 0.88, indicating that it correctly classifies 88.0% of the instances in the evaluation set. Based on this classification report, the model successfully distinguishes between overexposed and good instances, achieving high precision, recall, and F1-scores.

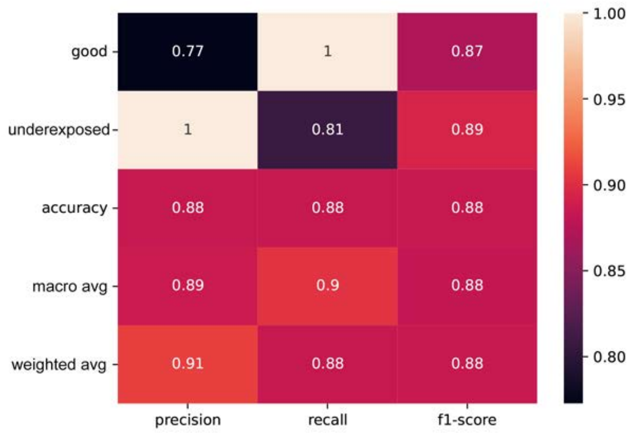


Figure 6.7: Underexposed Inanimate model classification report.

The evaluation results of the four exposure models suggest a mixed performance. While the accuracy scores for all models are relatively high, indicating good classification ability, the elevated loss values indicate room for improvement in terms of precision and consistency. The models' precision values vary across classes but generally accurately predict both overexposed and good instances. However, recall values show some variability, particularly in identifying instances of the "good" class. The F1-scores reflect a reasonable balance between precision and recall for both classes. Overall, the models exhibit promising performance but would benefit from further refinement to reduce loss values and enhance precision in predicting exposure levels.

6.2.3 Blurry and Skewed Horizon models

The blurry model's loss and accuracy evaluation results, presented in Table 6.4, indicate a mixed performance. With an accuracy score of 81.7%, the model demonstrates a relatively high level of overall correctness in classifying blurry instances. This suggests that the model can correctly identify and classify a significant portion of the blurry images in the dataset. However, the high loss score of 45.9% indicates a substantial deviation between predicted and true labels. This suggests that the model's predictions for blurry images could be more precise and consistent. Therefore, while the accuracy score is respectable, the elevated loss score implies a need to improve the model's ability to predict and classify blurry images accurately. Fine-tuning and optimization techniques could enhance the model's performance and reduce the loss score, leading to more accurate and reliable predictions.

The skewed horizon model exhibits poor performance, with a high loss score of 65.8% and a low accuracy score of 0.5%, as can be seen in Table 6.4. These metrics indicate that the model struggles to classify images accurately concerning the presence of a skewed horizon. The high loss score suggests that the model's predictions deviate significantly from the ground truth labels during training, indicating a lack of precision and consistency. The low accuracy score confirms that the model's predictions align with random chance, as it correctly classifies only 50.0% of the instances. This model requires significant improvements to identify and rectify images with skewed horizons effectively. Possible strategies for enhancement involve adjusting the model architecture, increasing the dataset size, or refining the training process to achieve better results.

Table 6.4: The loss and accuracy scores of the blurry and skewed horizon CNN model. Calculated using their respective validation datasets.

	Loss	**Accuracy**
Blurry	45.9%	81.7%
Skewed Horizon	65.8%	50.0%

The classification report for the *blurry model*, depicted in Figure 6.8, evaluates its performance in distinguishing between blurry and non-blurry images. The precision values for class 0 (blurry) and class 1 (good) are 0.81 and 0.76, respectively, indicating that when the model predicts a class, it is correct around 81.0% of the time for class 0 and 76.0% for class 1. The recall values, which measure the model's ability to identify instances of each class, are 0.59 for class 0 and 0.90 for class 1. This suggests that the model correctly identifies approximately 59.0% of blurry and 90.0% of non-blurry instances. The F1-scores, balancing precision and recall, are 0.68 for class 0 and 0.83 for class 1, indicating a reasonable trade-off between precision and recall.

Overall, the model achieves an accuracy of 0.77, correctly classifying 77.0% of the instances in the evaluation set. The average macro F1-score of 0.75 reflects a balanced performance across classes, while the weighted average F1-score of 0.77 suggests that the model's performance is consistent, considering the class distribution in the evaluation set. Based on this classification report, the model demonstrates a reasonable ability to classify images as blurry or non-blurry with moderate precision, recall, and F1-scores. However, there is room for improvement, particularly in enhancing the precision and recall for blurry instances, to further enhance the model's performance.

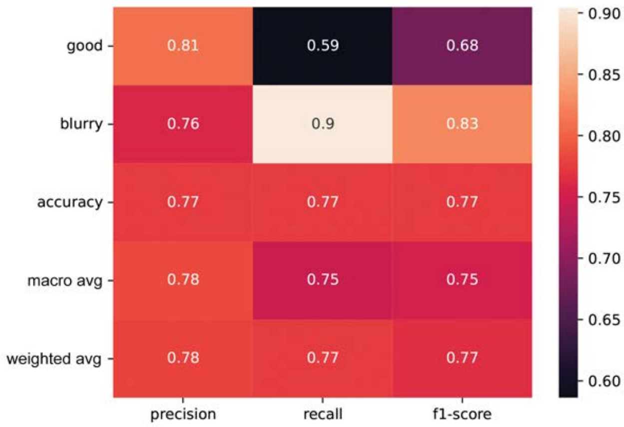


Figure 6.8: Blurry model classification report.

Figure 6.9 present the classification report for the *skewed horizon model*, and indicates moderate performance. The precision, recall, and F1-score for both classes, 0 (skewed-horizon) and 1(good), are relatively balanced but not very high. The model achieves a weighted average precision, recall, and F1-score of around 0.54, indicating that it is able to classify instances from both classes, but with a moderate level of accuracy. The overall accuracy of the model is 0.53, suggesting that it correctly predicts the class label for approximately 53.0% of the instances. However, there is room for improvement in terms of performance, as indicated by the relatively low scores across precision, recall, and F1-score. Further adjustments, such as refining the model architecture or increasing the dataset size, may be necessary to enhance its predictive capabilities and achieve better results.

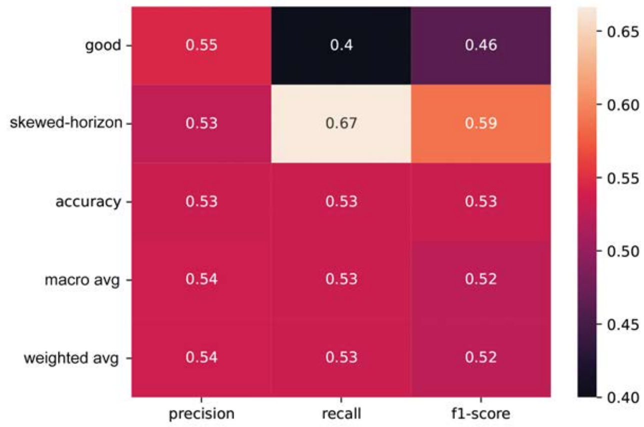


Figure 6.9: Skewed Horizon classification report.

6.3 Comparison Between Existing Software

The comparison table in Table 6.5 provides an overview of photo-culling software options discussed in Section 2.3, along with their key features. It directly compares them to DeepRoom and evaluates the incorporation of AI assistance, duplicate detection, subject identification, and landscape photography support. Additionally, compatibility with Adobe Lightroom, widely used among photographers, is considered, enabling seamless export of culled images and ratings to streamline post-production workflows.

Table 6.5: Photo culling software comparison. Collected from the capstone project "Determining the Relevance of Photographs: A Deep Learning Approach" written by Lintvedt, Victoria K. L. in 2022.

	DeepRoom	AfterShoot	Narrative Select	Photo Culling	Adobe Lightroom
Uses AI	*Yes*	Yes	Yes	Yes	No
Duplicate Detection	*No*	Yes	Yes	Yes	No
Subject Detection	*Yes*	Yes	Yes	Not stated	Yes
Designed for landscape	***Yes***	No	No	Not stated	No
Lightroom Compatible	*No*	Yes	Yes	Not stated	Yes

Among the AI-based systems, three offer similar functionalities primarily focused on assisting in culling photos with animate subjects. DeepRoom, although not currently deployed, possesses the underlying architecture for future development. Compared to other popular photo-culling software, DeepRoom lacks features like duplicate detection and detection of human blinks, indicating the need for further refinement to match the complexity and user-friendliness of existing options. Adobe Lightroom [64], while not utilizing AI for photo culling, is included in the comparison due to its widespread usage among photographers, particularly for photo editing with AI assistance.

None of the other applications explicitly specify optimization for culling inanimate objects, such as landscape photography, potentially limiting their effectiveness for photographers who frequently capture such images. This is where DeepRoom stands out by introducing something new. The models supporting the DeepRoom architecture are designed to analyze both animate and inanimate subjects, aiming to cater to the diverse repertoire of photographers.

6.4 Summary

The object detection model demonstrates promising results with high precision, recall, and mean average precision scores. These scores reflect the model's exceptional ability to detect and classify animals and humans in the dataset accurately. The model achieves remarkable precision in identifying and localizing objects, ensuring minimal false positive detection. With a high recall score, the model successfully captures the majority of instances of the target objects. Furthermore, the model's remarkably low loss values indicate its precision and optimization, producing accurate predictions that closely align with the labels. This optimized model enhances object detection performance and contributes to reliable and consistent results. These findings emphasize the model's proficiency in precise object detection, making it a valuable asset in various applications requiring accurate object recognition and localization.

The exposure models exhibit overall good accuracy scores, indicating their ability to correctly classify over- and underexposure with a reasonable level of accuracy. However, the high loss values suggest inconsistencies and imprecision in the exposure classification. One potential factor contributing to the high loss values could be underfitting, where the models fail to capture the underlying patterns and complexity of the images. Increasing the dataset size may provide more data for the models to learn from and better identify the underlying exposure patterns. Enhancing the models' capacity through increased complexity or additional layers/units and adjusting hyperparameters can improve their performance. Despite the need for further improvements, the classification reports demonstrate a reasonable balance between precision and recall for both classes on all exposure models.

The *underexposure inanimate* model stands out as the most robust among the models, with the highest accuracy score and the lowest loss score, being less than half of the rest. It also demonstrates the best results in the classification reports. The distinction in performance between this model and the other three models can be attributed to the curation of the datasets, as all models underwent identical preprocessing, model construction, and training processes. Since all four models were trained on datasets of the same size, the disparity in performance may stem from the quality of dataset curation. Although there is room for improvement, this model shows the most promising performance among the CNN models.

The performance of the *blurry model* could be more promising. While it demonstrates a relatively good accuracy score, the significantly high loss score is a cause for concern. This issue could be attributed to similar challenges encountered with underfitting, suggesting that the model's performance could be enhanced through similar interventions as discussed with the exposure models. Although the blurry model possesses a dataset twice the size of the other models, the quality of the dataset might be a contributing factor to its suboptimal performance, necessitating further improvements. Additionally, this model type may require a substantially larger dataset to discern the underlying patterns associated with blurry photos effectively.

The *skewed horizon* model's performance is significantly lower than the other models, evident from its high loss score exceeding the accuracy score of 50%. Despite several modifications made to the model's architecture and the curation of a more comprehensive dataset free of misleading images, the results obtained in this study remained suboptimal. Enhancing the model's performance requires addressing the inherent challenges associated with this classification task, such as training the CNN to recognize and distinguish between straight and skewed horizons. One potential avenue for improvement involves curating a larger dataset tailored explicitly for this task and subsequently fine-tuning the model. It is worth acknowledging that skewed horizon detection is a complex problem that may require additional refinements and strategies to achieve better results.

DeepRoom introduces a unique feature that sets it apart from existing software: the ability to rate inanimate photos. While it may lack certain features in other software, such as duplicate detection, it focuses on providing a versatile system that accommodates photographers of all genres. DeepRoom aims to be inclusive and serve the needs of photographers, regardless of the subject matter they are capturing.

7 Discussion

This chapter establishes a platform to acknowledge and engage with the observations encountered throughout the research process. By delving into these challenges, readers are granted a comprehensive understanding of the study's context and are presented with valuable insights into the potential limitations that emerged. Including this chapter greatly enhances the transparency and depth of the study, fostering a more robust and informative research experience.

All the photographs presented in this chapter have been captured by the author, underscoring the firsthand nature of the visual material. This ensures a direct connection between the research and the actual photographic examples.

7.1 Complexity of a Photograph

Defining what constitutes a good photograph and designing an effective rating system for photography presents a significant challenge in this thesis. The approach involved analyzing a photograph's composition and identifying potential flaws that may arise during the composition process, thereby offering a theoretical framework for addressing this question. Consequently, the rating system was developed based on key factors such as exposure, sharpness, focus, and skewed horizon in landscape photography. Additionally, graininess in photos could be considered since it is typically avoided in photography.

Determining the quality of a photograph is inherently subjective, as it relies on the photographer's perspective and the individual assessing the image. Technical aspects that might categorize a photograph as "bad" can also be intentionally employed to achieve a desired artistic outcome. Some of these artistic choices will be discussed in the following sections, and highlight how they challenge the current approach to photography rating.

7.2 Intentional Blurriness

Capturing intentional blur can be a deliberate artistic choice in photography, as demonstrated in Figure 7.1. Specific scenarios can benefit from intentionally introduced blur, like capturing the movement of water or a car. By employing techniques such as long exposures while maintaining camera stability, only the dynamic elements of the scene exhibit noticeable changes. This approach results in a smoother and more dynamic appearance for subjects like flowing water, as depicted in Figure 7.1a, or the streaks of light from a moving car, as depicted in Figure 7.1b. These images possess a distinct aesthetic appeal, showcasing the captivating effects of intentional blur.

(a) Long exposure of water. **(b)** Long exposure of car.

Figure 7.1: Photos illustrating using long exposure to create intentional blur.

However, when evaluating images with intentional blur, a challenge arises. A blurry model trained to identify sharpness and clarity might classify the intentional blur as undesirable blurriness, potentially resulting in a lower rating for these visually appealing photographs. It is necessary to address this challenge to ensure accurate assessments and ratings of images, accounting for the artistic choices and intentional blur effects photographers employ.

7.3 Night Photography and Intentional Underexposure

Night Photography and the Challenges of Underexposure Models Night photography introduces an intriguing aspect to consider when evaluating underexposure models. Images captured in low-light or nighttime settings often embrace the darkness and rely on available light sources to create captivating scenes, as illustrated in Figure 7.2a. These images possess a unique ambiance and can evoke a particular mood through the interplay of light and shadows in a dark environment.

One distinctive artistic element purposely uses underexposure, as shown in Figure 7.2b. Silhouettes are created when the main subject appears in a dark, featureless shape against a brighter background. This effect is achieved by intentionally underexposing the image, allowing the subject to appear as a silhouette against the backdrop of ambient or artificial light sources. It adds a powerful visual impact and can create a sense of mystery and drama within the photograph. The resulting silhouette is a powerful compositional element, drawing the viewer's attention to the subject and conveying a narrative or evoking emotions.

(a) Night photography **(b)** Silhouette

Figure 7.2: Photos illustrating night photography and silhouettes.

However, evaluating night photography, as well as the intentional use of underexposure, using underexposure models, presents particular challenges. Given the inherently darker nature of these images, the models may assign them a lower rating due to the underexposed appearance. This discrepancy arises because underexposure models prioritize brightness and exposure levels associated with well-lit scenes. As a result, such models may need to adequately capture or appreciate the artistic intentions and unique atmosphere conveyed by night photography. Addressing this challenge is crucial to ensure that the evaluation and rating of night photography accurately reflect its artistic value and the deliberate use of darkness and available light sources.

7.4 Presence of People in Landscape and Cityscape Photography

The object detection model developed in this thesis focuses on identifying animals and humans within images. However, it is crucial to recognize that specific photography scenarios involve the presence of humans without them being the primary subject of the composition. This is particularly evident in cityscape photography, where the intention is to capture the city's essence rather than emphasize individual passersby. An example of such a scenario is depicted in Figure 7.3. While the underlying architecture of the object detection model may successfully detect these individuals, it is essential to note that classifying these images solely as "animate" may not accurately reflect their intended purpose as land- or cityscape photographs.

Figure 7.3: Cityscape photography with people.

To address this challenge, incorporating adjustable settings or features within the model can offer more flexibility and customization. One potential approach involves introducing a user-adjustable threshold that determines the minimum proportion of human subjects required in an image to be classified as "animate." For example, users could set a threshold of 20% such that images containing less than that percentage of human subjects would be categorized as "inanimate" photographs. This approach allows for a more nuanced classification based on the presence and significance of human elements within the composition.

Additionally, implementing an override function could allow users to choose whether their dataset should be rated solely based on inanimate or animate factors. This functionality empowers users to align the classification and rating process with their specific requirements and preferences. By offering these options, the model can better accommodate diverse photography scenarios, ensuring that the classification accurately reflects each image's intended purpose and creative intent.

Addressing the nuances of human presence in non-human-centric compositions within photography enhances the versatility and applicability of the object detection model, making it more adaptable to various photographic styles and genres.

7.5 Summary

In delving deeper into the intricacies of creating a photography rating system, it becomes apparent that several factors warrant careful consideration. The subjective nature of what constitutes a "good" photograph and the broad spectrum of individual preferences highlight the impracticality of adopting a one-size-fits-all approach. Photography is a realm of boundless creativity and personal interpretation where diverse styles and artistic visions flourish.

In response to this complexity, DeepRoom was explicitly designed to prioritize the technical aspects of photography within its rating system. However, despite their potential artistic merit, the examples discussed in this chapter might have received lower scores within the DeepRoom architecture. This observation underscores the need for system adjustments to accommodate such compositions better and align with users' expectations.

One potential solution involves training specific models tailored to these unique scenarios, enabling users to select the most appropriate model for their photography datasets. Users gain heightened control over the rating process by tailoring the system to cater to different styles, genres, or artistic intentions. This approach empowers photographers to align the system's evaluation with their specific goals and preferences, leveraging the full potential of deep learning techniques to enhance their creative vision.

DeepRoom can evolve into a more versatile and adaptable tool by addressing the challenges of subjectivity and diversity in photography. By offering tailored models and user-controlled evaluation parameters, the system becomes better equipped to accommodate a broader range of compositions, embracing the rich tapestry of creative expression within photography.

8

Conclusions and Future Work

This thesis explores deep learning in imagery, explicitly focusing on photography evaluation. It investigates the design and implementation of a system that uses deep learning models to assess the technical criteria of photographs and determine their quality.

During this study, the DeepRoom system was developed as a deep learning-based rating system for photography. DeepRoom comprises seven deep learning models designed to assess and categorize photographs as good or bad. The rating system focuses on the technical aspects of image composition, considering factors like blurriness and exposure to objectively evaluate a photograph's quality. A key feature of DeepRoom is its ability to analyze both animate and inanimate photographs using object detection techniques to differentiate between them. The images are processed using specialized CNN models trained to evaluate specific features.

Most models in DeepRoom demonstrate strong performance with high accuracy and F1-scores, indicating their proficiency in correctly rating images. However, certain aspects of the images, such as blurriness and skewed horizon, proved more challenging to distinguish accurately, resulting in higher loss values. This suggests that further improvements are necessary to enhance the models' ability to handle these complex features and improve their performance.

The findings of this research demonstrate the potential of deep learning in accurately rating and evaluating photographs. However, the study also acknowledges the subjectivity of photography and the limitations of any automated system in fully capturing the fine qualities of a photograph. To address this, the thesis proposes the inclusion of adjustable settings and customizable options in the rating system, allowing users to align the evaluation process with their specific requirements and preferences.

In conclusion, this thesis provides valuable insights into using deep learning in photography evaluation. It addresses the research problem by designing and implementing a system that effectively rates and evaluates photographs based on photography-specific technical criteria. This research contributes to deep learning and photography, opening up new avenues for objectively and consistently assessing photographs.

8.1 Future Work

The DeepRoom system, currently in its prototype stage, still requires significant development and optimization to reach its full potential. The following section will outline the key areas of future work that aim to enhance and optimize the system's performance and capabilities.

Optimization

The CNN models used in DeepRoom should be optimized and trained further, particularly for those with high loss values. This can be achieved by enlarging the datasets used for training to cover a broader range of scenarios and image compositions. By incorporating more diverse images, the models can better learn to handle challenging features such as blurriness and skewed horizons, improving their rating accuracy.

Additionally, it is recommended to analyze and reconstruct the architectures of the blurry and skewed horizon models. This analysis may involve exploring different network architectures, adjusting hyperparameters, and incorporating specific image-processing techniques to address the complexities associated with blurriness and skewed horizons. More accurate results can be achieved by iteratively refining the models, enhancing DeepRoom's ability to rate photographs with these challenging features effectively.

Additional features

The current implementation of DeepRoom serves as a foundational prototype, meeting the minimum system requirements. However, several additional features should be considered to optimize the system further. Firstly, handling large datasets efficiently and ensuring scalability is crucial for enhanced performance. The system should be designed to process and manage substantial numbers of photos effectively.

Another important feature is the detection and grouping of duplicates, enabling the system to identify and group similar photos together, even if they are not exact duplicates. This functionality simplifies the analysis and rating process by allowing users to conveniently compare and evaluate similar images. Additionally, personalized culling preferences should be implemented, empowering users to customize their photo culling process according to their preferences. This can include options such as displaying only the worst-rated photos for deletion or showcasing the highest-rated photos for further post-production.

Deployment

DeepRoom is currently in the architectural stage, with the rating models being the primary focus. To move forward, the next step is to implement a deployment system that integrates and combines these models into a cohesive and user-friendly application. One possible approach is to develop a console-based system, providing a straightforward and efficient interface for users to interact with DeepRoom.

The console-based system would involve designing a command-line interface (CLI) that allows users to input commands and receive output directly in the console. This approach offers simplicity and ease of use, particularly in server or non-graphical operating systems where a GUI may not be available or necessary. Alternatively, a fully fleshed front-end application with a graphical user interface (GUI) could be developed, offering a more visually appealing and intuitive user experience. The choice between a console-based system and a GUI application depends on user preferences, platform compatibility, and the level of functionality and interactivity desired.

Trainable models

The thesis has explored what constitutes a good photograph, with DeepRoom adopting an objective and technical approach in its rating system. However, a different approach that could enhance the system is incorporating trainable models, enabling users to customize the system according to their preferences. The initial model provided by DeepRoom serves as a baseline, incorporating the current rating criteria. Users can evaluate and train these models to align with their unique photography style.

This empowers users to fine-tune the rating system, creating a tailored experience that aligns with their specific criteria. By allowing users to review and adjust the system's rating results, they can directly influence the ratings assigned to photos, providing feedback to the DL models. Over time, the models will adapt to incorporate the user's ratings, gradually refining the predefined rating system.

A

Deep Learning Model Training

This appendix visually represents the training processes for each of the six CNN models and the object detection training graphs. These graphs offer insights into the training progress and performance of the models, showcasing their learning curves and improvements over time.

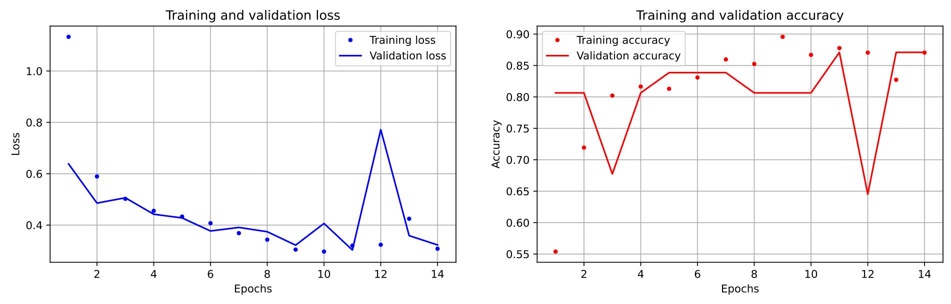


Figure A.1: Overexposed Animate training graph.

In Figure A.1, the overexposed animate CNN model underwent 14 epochs. The loss values were similar between the training and validation datasets, with a notable spike in the validation set during the twelfth epoch. The accuracy scores varied slightly between the two datasets, with the validation dataset performing slightly worse overall than the training dataset.

Figure A.2: Overexposed Inanimate training graph.

Figure A.2 displays the overexposed inanimate CNN model's eleven epochs. The training and validation datasets exhibited similar varying loss values, with the validation dataset showing lower losses initially but spiking towards the end of training. Accuracy scores remained consistent between the datasets.

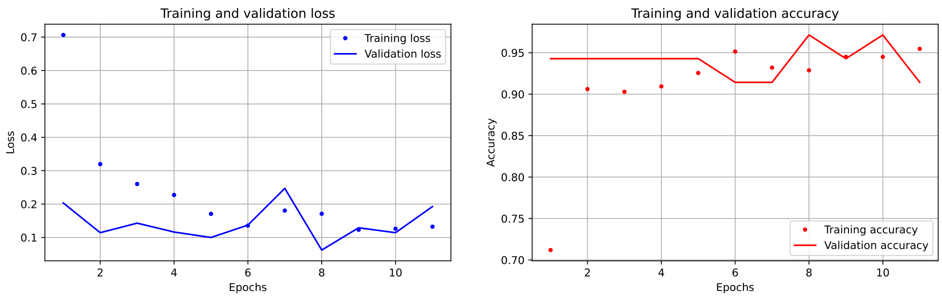


Figure A.3: Overexposed Animate training graph.

Figure A.3 depicts the underexposed animate CNN model's eleven epochs. The training and validation datasets maintained similar loss and accuracy values consistently during the training.

Figure A.4: Overexposed Inanimate training graph.

Figure A.4 illustrates the eleven epochs of the underexposed inanimate CNN model. The training and validation datasets exhibit similar curves for loss and accuracy values throughout training. However, the validation dataset shows slightly poorer performance with higher loss values and lower accuracy compared to the training dataset.

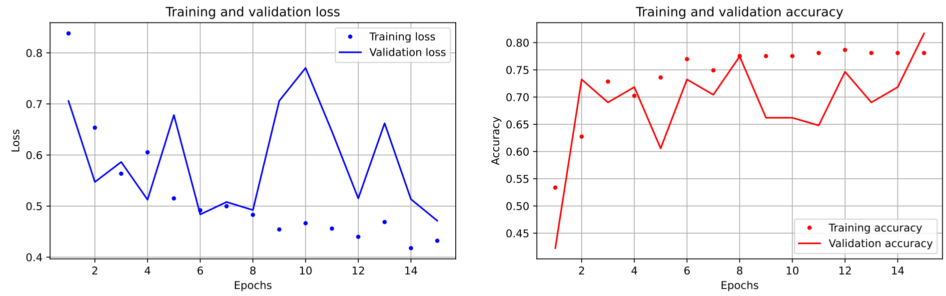


Figure A.5: Blurry model training graph.

Figure A.5 displays the fifteen epochs of the blurry model. The training dataset exhibits a smooth decrease in loss values and a consistent increase in accuracy. However, the validation dataset shows noticeable spikes in both loss and accuracy during training before converging to values similar to the training dataset.

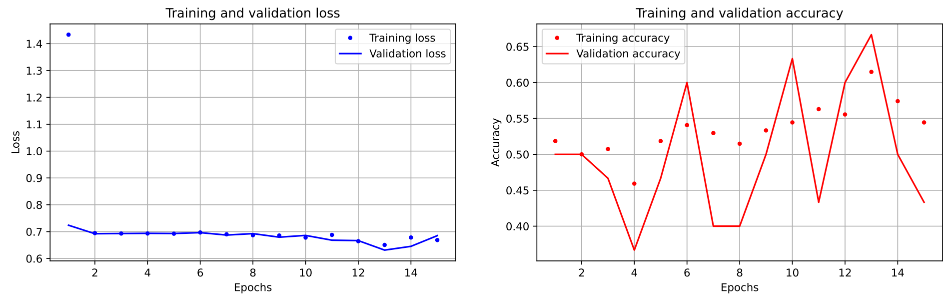


Figure A.6: Skewed Horizon model training graph.

Figure A.6 shows the skewed horizon CNN model training. The loss values remain consistent, while the accuracy scores fluctuate with noticeable spikes, particularly in the validation dataset.

Bibliography

[1] Oracle Cloud Infrastructure (OCI). What is ai? learn about artificial intelligence. https://www.oracle.com/artificial-intelligence/what-is-ai/, 2022. Retrieved: 2023-01-15.

[2] Liu, J., Kong, X., Xia, F., Bai, X., Wang, L., Qing, Q., & Lee, I. Artificial intelligence in the 21st century. *IEEE Access*, 6:34403–34421, 2018. http://dx.doi.org/10.1109/ACCESS.2018.2819688.

[3] McKinsey & Company. The state of ai in 2022—and a half decade in review. https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-in-2022-and-a-half-decade-in-review, 2022. Retrieved: 2023-05-01.

[4] Open AI. Introducing chatgpt. https://openai.com/blog/chatgpt, 2022. Retrieved: 2023-05-01.

[5] Hu, K. Chatgpt sets record for fastest-growing user base - analyst note. https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/, 2023. Retrieved: 2023-05-01.

[6] Tesla. About us. https://www.tesla.com/about, 2023. Retrieved: 2023-05-01.

[7] Tesla. Ai robotics; neural networks. https://www.tesla.com/AI, 2023. Retrieved: 2023-05-01.

[8] Anyoha, R. *The History of Artificial Intelligence; Artificial Intelligence is Everywhere.* harvard university, the graduate school of arts and sciences. https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/, 2017. Retrieved: 2023-05-01.

[9] Haenlein, M. & Kaplan, A. A brief history of artificial intelligence: On the past, present, and future of artificial intelligence. *California Management*

Review, 61(4):5–14, 2019. https://doi.org/10.1177/0008125619864925.

[10] Ongsulee, P. Artificial intelligence, machine learning and deep learning. *2017 15th International Conference on ICT and Knowledge Engineering (ICT&KE)*, pages 1–6, 2017. https://doi.org/10.1109/ICTKE.2017.8259629.

[11] IBM Cloud Education. *Neural Networks*. https://www.ibm.com/cloud/learn/neural-networks, 2020. Retrieved: 2023-01-29.

[12] Hakansson, A. & Hartung, R. *Artificial Intelligence - Concepts, Areas, Techniques and Applications*. Studentlitteratur, 2020. Page 241.

[13] Mihajlovic, I. Everything you ever wanted to know about computer vision. https://towardsdatascience.com/everything-you-ever-wanted-to-know-about-computer-vision-heres-a-look-why-it-s-so-awesome-e8a58dfb641e, 2019. Retrieved: 2023-02-17.

[14] IBM. What is computer vision? https://www.ibm.com/topics/computer-vision?mhsrc=ibmsearch_a&mhq=computer%20vision, 2019. Retrieved: 2023-02-17.

[15] National Science and Media Museum. The first digital photos, from victorian technology to the internet. https://www.scienceandmediamuseum.org.uk/objects-and-stories/first-digital-photos, 2020. Retrieved: 2023-01-08.

[16] GlobalData Thematic Intelligence. History of computer vision: Timeline. https://www.verdict.co.uk/computer-vision-timeline/, 2020. Retrieved: 2023-05-01.

[17] Štetić, F. The history of computer vision and the evolution of autonomous vehicles. https://www.megatrend.com/en/the-history-of-computer-vision-and-the-evolution-of-autonomous-vehicles/, 2022. Retrieved: 2023-05-01.

[18] SAS Institute. Computer vision: What it is and why it matters. https://www.sas.com/en_us/insights/analytics/computer-vision.html, 2022. Retrieved: 2023-03-17.

[19] Clickworker. Computer vision. https://www.clickworker.com/ai-glossary/computer-vision/, 2022. Retrieved: 2023-05-01.

[20] Mihajlovic, I. How artificial intelligence is impacting our ev-

eryday lives; and how you already encounter it every day. https://towardsdatascience.com/how-artificial-intelligence-is-impacting-our-everyday-lives-eae3b63379e1, 2019. Retrieved: 2023-05-01.

[21] Kiniulis, K. 12 mobile photography statistics and how it's changing our world. https://www.eksposure.com/mobile-photography-statistics/, 2022. Retrieved: 2023-05-01.

[22] Instagram. Instagram [computer software]. https://www.instagram.com/, 2023. Retrieved: 2023-03-16.

[23] Facebook. Facebook [computer software]. https://www.facebook.com/, 2023. Retrieved: 2022-03-16.

[24] Amon, M.J., Hasan, R., Hugenberg, K., Bertenthal, B., & Kapadia, A. Influencing photo sharing decisions on social media: A case of paradoxical findings. *Published in 2020 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA*, 2020. https://doi.org/10.1109/SP40000.2020.00006.

[25] Bartosz, K. Information overload: The real problem or a temporary phenomenon of modern cultures? *Themenschwerpunkt "Leben nach Zahlen. Digitale Kulturen und Datafizierung*", (14):1–10, 2022. https://doi.org/10.25598/JKM/2022-14.37.

[26] Håkansson, A. Portal of research methods and methodologies for research projects and degree projects. *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS)*, pages 1–7, 2013.

[27] Lintvedt, V. K. L. Determining the relevance of photographs: A deep learning approach. *University of Tromsø, The Arctic University of Norway*, 2022. Capstone project.

[28] Berndtsson, M., Hansson, J., Olsson, B., Lundell, B. *Thesis Projects: A Guide for Students in Computer Science and Information Systems*. 2008. p 12-14. https://doi.org/10.1007/978-1-84800-009-4.

[29] IBM Cloud Education. *Deep Learning*. https://www.ibm.com/cloud/learn/deep-learning, 2020. Retrieved: 2023-01-29.

[30] Alkabbani, H., Ahmadian, A., Zhu, Q., & Elkamel, A. Machine learning and metaheuristic methods for renewable power forecasting: A recent

review. *Frontiers in Chemical Engineering*, 3:665415, 2021. https://doi.org/10.3389/fceng.2021.665415.

[31] AI Wiki. Weights and biases. https://machine-learning.paperspace.com/wiki/weights-and-biases, 2022.

[32] Malik, F. Neural networks bias and weights. https://medium.com/fintechexplained/neural-networks-bias-and-weights-10b53e6285da, 2019. Retrieved: 2023-02-29.

[33] Hakansson, A. & Hartung, R. *Artificial Intelligence - Concepts, Areas, Techniques and Applications*. Studentlitteratur, 2020. Page 224.

[34] Hakansson, A. & Hartung, R. *Artificial Intelligence - Concepts, Areas, Techniques and Applications*. Studentlitteratur, 2020. Page 229.

[35] Hakansson, A. & Hartung, R. *Artificial Intelligence - Concepts, Areas, Techniques and Applications*. Studentlitteratur, 2020. Pages 248-252.

[36] Bandyopadhyay, H. An introduction to image segmentation: Deep learning vs. traditional [+examples]. v7 labs. https://www.v7labs.com/blog/image-segmentation-guide, 2021. Retrieved: 2023-02-17.

[37] Van Veen, F. The neural network zoo. https://www.asimovinstitute.org/neural-network-zoo/, 2016. Retrieved: 2023-05-17.

[38] Boesch, G. A complete guide to image classification in 2023. https://viso.ai/computer-vision/image-classification/, 2023. Retrieved: 2023-02-17.

[39] Sharma, P. Image classification vs. object detection vs. image segmentation. https://medium.com/analytics-vidhya/image-classification-vs-object-detection-vs-image-segmentation-f36db85fe81, 2019. Retrieved: 2023-02-17.

[40] Telus International. Your complete guide to image segmentation. https://www.telusinternational.com/articles/guide-to-image-segmentation, 2022. Retrieved: 2023-02-17.

[41] Kundu, R. Yolo: Algorithm for object detection explained [+examples]. https://www.v7labs.com/blog/yolo-object-detection, 2023. Retrieved: 2023-05-18.

[42] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. You only look once:

Unified, real-time object detection, 2016. *Cornell University*. https://doi.org/10.48550/arXiv.1506.02640.

[43] Karimi, G. Introduction to yolo algorithm for object detection. https://www.section.io/engineering-education/introduction-to-yolo-algorithm-for-object-detection/, 2021. Retrieved: 2023-05-18.

[44] Fairfax, S. What is the exposure triangle? https://www.polarprofilters.com/blogs/polarpro/the-three-elements-of-the-exposure-triangle, 2022. PolarPro. Retrieved: 2023-03-03.

[45] Lucid Vision Labs. Understanding the digital image sensor. https://thinklucid.com/tech-briefs/understanding-digital-image-sensors/, 2022. Retrieved: 2023-03-03.

[46] Naryškin, R. Underexposure vs overexposure – a beginner's guide. https://photographylife.com/underexposure-and-overexposure-in-photography, 2020. Retrieved: 2023-02-03.

[47] Foto Review. File formats for digital photos. https://www.photoreview.com.au/tips/editing/file-formats-for-digital-photos/, 2022. Retrieved: 2023-03-03.

[48] Kedves, T. How to fix an overexposed photo. https://expertphotography.com/fix-overexposed-photo/, 2022. Retrieved: 2023-01-25.

[49] Nichols, J.-L. 10 types of image file extensions and when to use them. https://blog.hubspot.com/insiders/different-types-of-image-files, 2023. Retrieved: 2023-03-25.

[50] Bond, S. *Jpeg vs RAW vs TIFF – What's the Difference? (File Formats). Expert Photography*. https://expertphotography.com/tiff-vs-jpeg-vs-raw/, 2022. Retrieved: 2023-02-25.

[51] Ogonek, C. Is shooting raw or jpeg better for you? https://www.nikonusa.com/en/learn-and-explore/a/tips-and-techniques/is-shooting-raw-or-jpeg-better-for-you.html, 2022. Retrieved: 2023-02-25.

[52] Amin, E. Photography post production a complete guide. https://ehabphotography.com/photography-post-production-a-complete-guide/. Retrieved: 2023-02-25.

[53] MapSystems. Photo culling: What it means and why is it im-

portant? https://mapsystemsindia.com/resources/what-is-photo-culling.html. Retrieved: 2023-02-25.

[54] Kivowitz, S. W. A complete guide to culling photographs. https://www.imagen-ai.com/post/culling-photographs-guide, 2022. Retrieved: 2022-10-25.

[55] Williams, B. Photo editing vs photo manipulation - what's the difference. https://www.bwillcreative.com/photo-editing-vs-photo-manipulation/, 2020. Retrieved: 2023-03-25.

[56] AfterShoot. Aftershoot [computer software]. https://aftershoot.com/, 2022. Retrieved: 2023-02-18.

[57] Dwivedi, H. How does aftershoot's ai work? https://support.aftershoot.com/en/articles/5353891-how-does-aftershoot-s-ai-work. Retrieved: 2023-02-24.

[58] Benson, J. What if the ai misses out on a creative image that i want? https://support.aftershoot.com/en/articles/5353892-what-if-the-ai-misses-out-on-a-creative-image-that-i-want, 2022. Retrieved: 2023-02-24.

[59] AfterShoot. Faqs. https://aftershoot.com/culling-faq/, 2022. Retrieved: 2023-05-24.

[60] Narrative. Narrative select [computer software]. https://narrative.so/select, 2022. Retrieved: 2023-02-18.

[61] Narrative Select. Select advanced learning. https://narrative.so/select/lessons/advanced-learning, 2020. Retrieved: 2023-02-24.

[62] Narrative Select. Select quickstart tutorial. https://narrative.so/select/lessons/quickstart-guide, 2020. Retrieved: 2023-02-24.

[63] Canon. *Need A Digital Photo Assistant? Canon U.S.A. Has You Covered With The Company's First-Ever Photo Culling App.* inc. canon u.s.a. https://www.usa.canon.com/newsroom/2021/20210211-app, 2021. Retrieved: 2023-02-25.

[64] Adobe. Adobe lightroom photoshop (version 6) [computer software]. https://tinyurl.com/AdobeLightroomPhotoshop, 2022. Retrieved: 2023-02-18.

[65] Condon, M. Best photography culling software to speed up your image selection workflow. https://shotkit.com/best-photo-culling-software/, 2023. Retrieved: 2023-02-25.

[66] Dam, P. 5 tips to cull images in lightroom. https://www.adorama.com/alc/5-tips-to-cull-images-in-lightroom/, 2021.

[67] Question Pro. Exploratory research: Definition, types and characteristics. https://www.questionpro.com/blog/exploratory-research/, 2020. Retrieved: 2023-04-26.

[68] Anaconda Inc. Anaconda software distribution [computer software]. https://docs.anaconda.com/, 2020. Vers. 2-2.4.0. Anaconda Documentation.

[69] Kluyver, T., Ragan-Kelley, B., Perez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., & Willing, S. Jupyter notebooks – a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.

[70] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Va sudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X. TensorFlow [Computer Software]: Large-scale machine learning on heterogeneous systems. 2015. Available from tensorflow.org.

[71] Chollet, F. Keras [computer software]. https://keras.io, 2015.

[72] Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. https://doi.org/10.1109/MCSE.2007.55.

[73] Waskom, M. L. Seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. https://doi.org/10.21105/joss.03021.

[74] Dwyer, B., Nelson, J. (2022), Solawetz, J., et. al. Roboflow (version 1.0) [computer software], 2022.

[75] Anaconda. Anaconda [computer software]. we're not just a company; we're a movement. `https://www.anaconda.com/`. Retrieved: 2023-05-24.

[76] AMD. Amd ryzen™ 7 3700x: Specifications [computer hardware]. `https://www.amd.com/en/product/8446`. Retrieved: 2023-05-24.

[77] Nvidia. Geforce gtx 1070 | specifications | geforce [computer hardware]. `https://www.nvidia.com/en-gb/geforce/graphics-cards/geforce-gtx-1070/specifications/`, 2022. Retrieved: 2023-05-24.

[78] Henderson, P. & Ferrari, V. End-to-end training of object class detectors for mean average precision. *Computer Vision - ACCV 2016. Springer International Publishing*, pages 198–213, 2017. `https://doi.org/10.48550/arXiv.1607.03476`.

[79] Ambolt AI. How object detectors learn. `https://ambolt.io/en/how-object-detectors-learn/`. Retrieved: 2023-05-28.

[80] Tharwat, A. Classification assessment methods. *Applied Computing and Informatics*, 17(1):168–192, 2021. `https://doi.org/10.1016/j.aci.2018.08.003`.

[81] Powers, D. Evaluation: From precision, recall and f-factor to roc, informedness, markedness correlation. *Machine Learning: Science and Technology*, 2, 2008.

[82] Jurman, G. & Chicco, D. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1):6, 2020. `https://doi.org/10.1186/s12864-019-6413-7`.

[83] Brownlee, J. What is the difference between a batch and an epoch in a neural network. *Machine Learning Mastery*, 20, 2018.

[84] N., Siddesh, C. S. & Kuldeep, S Kukreja, H., Bharath. An introduction to artificial neural network. *Int J Adv Res Innov Ideas Educ*, 1:27–30, 2016.