IMAGE QUALITY AND BEAUTY CLASSIFICATION USING DEEP LEARNING

by

Arash Golchubian

A Dissertation Submitted to the Faculty of

The College of Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

Florida Atlantic University

Boca Raton, FL

July 2022

Copyright 2022 by Arash Golchubian

IMAGE QUALITY AND BEAUTY CLASSIFICATION USING DEEP LEARNING

by

Arash Golchubian

This dissertation was prepared under the direction of the candidate's dissertation advisor, Dr. Mehrdad Nojoumian, Department of Electrical Engineering and Computer Science, and has been approved by the members of his supervisory committee. It was submitted to the faculty of the College of Engineering and Computer Science and was accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

SUPERVISORY COMMITTEE:

Mehrdad Nojoumian (Aug 2, 2022 15:23 EDT)

Mehrdad Nojoumian, Ph.D.
Dissertation Advisor

Borko Furht (Aug 5, 2022 15:14 EDT)

Borko Furht, Ph.D.

Taghi Khoshgoftaar
Taghi Khoshgoftaar (Aug 5, 2022 15:16 EDT)

Taghi Khoshgoftaar, Ph.D.

Oge Marques (Aug 5, 2022 16:50 EDT)

Oge Marques, Ph.D.

Hanqi Zhuang, Ph.D.
Chair, Department of Electrical Engineering and Computer Science

Stella N. Batalama, Ph.D.
Dean, The College of Engineering and Computer Science

Robert W. Stackman, Jr., Ph.D.
Dean, Graduate College

Date

ACKNOWLEDGEMENTS

I would first and foremost like to express my deepest gratitude to my dissertation advisor Dr. Mehrdad Nojoumian whose support and insightful counsel has guided me through my doctoral studies at Florida Atlantic University. I am grateful for all of the hours Mehrdad has spent on this dissertation, for helping me through the difficult challenges of my research, and for always supporting my research effort with encouragement and helpful nudges in the right direction. I am forever in his debt for everything he has taught me.

I would like to take this opportunity to send my sincere thanks to members of my dissertation committee: Dr. Borko Furht, Dr. Taghi Khoshgoftaar, and Dr. Oge Marques, for the sage advice and guidance throughout my PhD studies.

I would also like to extend my appreciation to all of the teachers and professors who have inspired and encouraged me to continue to be curious. I will forever be grateful to them.

I would like to give my thanks to the wonderful staff of Florida Atlantic University for guiding me through the process and procedures of the university. Your friendly attitude and general willingness to help, made my experience an enjoyable and positive memory. A big thank you goes out to Jean Mangiaracina and the entire department staff. I would not have been able to keep up with all of the deadlines and procedures without your constant support.

I would like to thank my family and friends, for always showing interest in my studies and providing much needed motivation. I would especially like to thank my amazing wife, Sanaz Imen, for supporting me and for encouraging me to continue my education. She truly is my inspiration.

ABSTRACT

Author:	Arash Golchubian
Title:	Image Quality and Beauty Classification Using Deep Learning
Institution:	Florida Atlantic University
Dissertation Advisor:	Dr. Mehrdad Nojoumian
Degree:	Doctor of Philosophy
Year:	2022

The field of computer vision has grown by leaps and bounds in the past decade. The rapid advances can be largely attributed to advances made in the field of Artificial Neural Networks and more specifically can be attributed to the rapid advancement of Convolutional Neural Networks (CNN) and Deep Learning. One area that is of great interest to the research community at large is the ability to detect the quality of images in the sense of technical parameters such as blurriness, encoding artifacts, saturation, and lighting, as well as for its' aesthetic appeal. The purpose of such a mechanism could be detecting and discarding noisy, blurry, dark, or over exposed images, as well as detecting images that would be considered beautiful by a majority of viewers. In this dissertation, the detection of various quality and aesthetic aspects of an image using CNNs is explored. This research produced two datasets that are manually labeled for quality issues such as blur, poor lighting, and digital noise, and for their aesthetic qualities, and Convolutional Neural Networks were designed and trained using these datasets. Lastly, two case studies were performed to show the real-world impact of this research to traffic sign detection and medical image diagnosis.

To:

My loving wife, Sanaz.

IMAGE QUALITY AND BEAUTY CLASSIFICATION USING DEEP LEARNING

List of Tables x

List of Figures xi

1 Introduction 1

1.1 Overview 1

1.2 Motivations and Contributions 2

1.2.1 Datasets 2

1.2.2 Detection of Image Quality 6

1.2.3 Detection of Aesthetic Quality 6

1.3 Outline 7

2 Preliminary Technical Materials 8

2.1 Machine Learning 8

2.2 Convolutional Neural Networks 8

2.2.1 Convolution Operation 9

2.2.2 Filtering 9

2.2.3 Optimizers 10

2.3 ResNet 13

2.4 Interrater Agreement 13

3 Literature Review 17

3.1 Methods for Classification of Photo Quality Parameters 17

3.1.1 Learning Based Approaches 17

3.2 Aesthetic Ranking of Photographs 19
3.3 Medical Diagnosis Classification 26

4 Photo Quality Classification Using Deep Learning 28
4.1 Introduction 28
4.2 Preliminary Materials 30
4.2.1 Image Quality Issues 30
4.3 Methodology 32
4.3.1 Convolutional Neural Networks 32
4.3.2 TensorFlow 34
4.3.3 Our Convolutional Neural Network 34
4.3.4 Our Implementation 36
4.4 Results 39
4.4.1 Additional Analysis 41
4.5 Application to Traffic Images 43
4.5.1 Initial Results 43
4.5.2 Transfer Learning 44
4.6 Conclusions and Future Directions 45

5 The Effects of Image Quality on Deep Learning Classification Performance 47
5.1 Introduction 47
5.2 Methodology 49
5.2.1 Data 49
5.2.2 Our Model 50
5.2.3 Training 50
5.3 Error Insertion Case Study 52
5.4 Results 53
5.4.1 Dataset 1 53

5.4.2 Dataset 2 55
5.4.3 Conclusions 56

6 Aesthetic Image Quality Ranking Using Deep Learning 57
6.1 Introduction 57
6.2 Methodology 60
6.2.1 Image Dataset 60
6.2.2 Rating Images 60
6.2.3 Our Model 62
6.2.4 Training 63
6.3 Results 64
6.4 Discussion and Conclusions 64

7 Closing Remarks and Future Works 69

LIST OF TABLES

2.1 Error rates (%, 10-crop testing) on ImageNet validation. All data was presented by He et al. [7]. 16

4.1 Layers and Parameters in Proposed CNN 37

4.2 Classifier Performance by Class 42

6.1 Number of images by deviation of model rating from human rating . 64

LIST OF FIGURES

1.1 White Noise 4

1.2 Clear Photo 2 5

2.1 The fundamental residual block showing the shortcut connection and activation layers [7] 14

3.1 JPEG 2000 Compression Artifacts 20

4.1 Sample Gaussian Blur Images 31

4.2 Dark/Bad Lighting 34

4.3 Convolutional Neural Network Architecture 35

4.4 60/20/20 Training Accuracy History 38

4.5 80/10/10 Training Accuracy History 39

4.6 80/10/10 Split 39

4.7 60/20/20 Split 39

4.8 Confusion Matrix - Traffic Images 40

4.9 Incorrectly Classified Reference Images 44

5.1 Image resizing process 49

5.2 Confusion matrix depicting the performance of the classifier. 51

5.3 Confusion matrix depicting the performance of the classifier after the test images have been injected with noise. 53

5.4 Confusion matrix depicting the performance of the classifier. 54

5.5 Confusion matrix depicting the performance of the classifier after the test images have been injected with noise. 55

6.1 Number of times each rating from 0-4 was selected 61

6.2 Number of images with $AD_{M(j)}$ obtained from human rankings, rounded to the nearest 0.1 66

6.3 Convolutional Neural Network Architecture 67

6.4 (A) Human ranking = 0, Model Ranking = 1; (B) Human ranking = 3, Model Ranking = 2; (c) Human ranking = 2, Model Ranking = 3; (D) Human ranking = 3, Model Ranking = 3 68

ACRONYMS

AI Artificial Intelligence. 2, 47, 64, 65

ANN Artificial Neural Network. 1, 8

CNN Convolutional Neural Network. 1, 2, 8, 9, 17, 20, 21, 26, 28, 48, 59, 60, 62, 65, 69

GAN Generative Adversarial Network. 27

GPU Graphics Processing Unit. 1, 62

ISIC International Skin Imaging Collaboration. 5, 49

ML Machine Learning. 2, 8

ResNet Residual Netowork. 13, 62

SGD Stochastic Gradient Descent. 50, 63

SVM Support Vector Machine. 8, 27

CHAPTER 1
INTRODUCTION

In this dissertation two Convolutional Neural Networks (CNNs) are proposed to 1) classify images by quality problems, and 2) rank images based on holistic aesthetic qualities. A case study is also performed to examine the effects of photo quality issues on medical image diagnosis using deep learning. This chapter provides a brief introduction to the topics, the motivations and contributions of this research, and the outline of this dissertation.

1.1 OVERVIEW

The field of computer vision has grown by leaps and bounds in the past decade. The rapid advances can be largely attributed to advances made in the field of Artificial Neural Networks (ANNs) and CNNs in particular. On the 30th of September in 2012, the winning CNN at the ImageNet Challenge was able to achieve a top-5 error rate of just 15.3%, which at the time was more than 10% better than the runner up. This was a turning point for deep learning, and a revolution was sparked. Made possible by the use of Graphics Processing Units (GPUs), CNNs which were conceptualized more than 3 decades earlier, took the machine learning and computer vision world by storm. Within just a few years, the top-5 error rate for the ImageNet challenge would drop to below 5% making classification of images by a computer more accessible than ever. Even with these advances, there are still large challenges remaining within this field. One area that is of great interest to the community at large is the ability to detect the quality of images in the sense of technical parameters such as blurriness, encoding

artifacts, saturation, and lighting, as well as for its' aesthetic appeal. The purpose of such a mechanism could be for the purposes of detecting and discarding noisy, blurry, dark, or over exposed images, or for the purposes of detecting images that would be considered beautiful by a majority of viewers. In this dissertation, the detection of various quality and aesthetic aspects of an image using CNNs is explored. This research produced two datasets which are manually labeled for quality issues such as blur, poor lighting, and digital noise, and for their aesthetic qualities. In addition, three purpose built CNNs are designed and trained achieving high accuracy. Lastly, two case studies were performed to show the real world impact of this research.

1.2 MOTIVATIONS AND CONTRIBUTIONS

The contributions and novelty of this research is 1) an end-to-end Machine Learning (ML) approach to detecting quality problems in images [1], 2) a case study exploring the effects of photo quality on medical image diagnosis [2] and 2) a unique approach for aesthetic ranking of an image using a novel CNN [3]. The motivation for this work is, 1) to allow for ML to better understand the quality of inputs that are being presented, 2) to use that insight to improve the overall accuracy of other tasks for which the ML algorithms are being used, and 3) to enable Artificial Intelligence (AI) to understand the holistic beauty of images so that we may use this system to build systems that can provide a more insightful experience for users.

1.2.1 Datasets

Three datasets have been curated during the course of this research. The datasets were built by collecting images from various online sources, processing, and manually labeling the images.

Photo Quality Dataset

To train a model that could be generalized for a large number types of photos, there was a need for a diverse dataset containing images of different subjects and taken in various lighting conditions. Since such a dataset was not readily available, a new dataset was created from the combination of images from the dataset created by Tang, Luo, and Wang [4] and a set of blurry images that were created specifically for this study. The blurry images were created using a Sony Alpha 6000 camera by manually defocussing the camera. The aperture, timing, and ISO were set automatically by the camera and vary for each image. The images were then resized prior to processing through python for performance reasons. The dataset contains a total of 125 out of focus images and 125 high quality images as classified in the prior publication [4]. Motion blur images were produced by using the OpenCV Python library and performing a motion blur operation on the reference images.

The Reference images were taken from the images labeled as high quality by Tang, Luo, and Wang [4] which were selected to ensure clarity and lack of blurry elements. Note that some of these images may contain out of focus or motion blurred sections, but those types of blurriness have been judged to be desirable. There are also images which are from the dataset by Sheikh et al. [5] which where modified to produce the JPEG 2000 and white noise images.

Aesthetic Ranking Dataset

The image dataset used to rate an image's aesthetic qualities was created by collecting photos from the websites flickr.com, unsplash.com, and pixabay.com. These platforms enable users to upload images to the site and provide the option to allow others to download their images. Images containing forests, bodies of water, and meadows were downloaded for this study. 2,000 images matching that criteria were downloaded at their original size.



Figure 1.1: White Noise



Figure 1.2: Clear Photo 2

Medical Image Dataset

The International Skin Imaging Collaboration (ISIC) is a partnership between academia and industry with the aim to facilitate the application of digital skin imaging. The archives contain thousands of labeled images of skin regions with various confirmed diagnosis. To conduct this study, we collected 3,408 images with a diagnosis of basal cell carcinoma and 3,408 images within one nine diagnosis groups. These groups include actinic keratosis, dermatofibroma, melanoma, nevus, pigmented benign keratosis, seborrheic keratosis, seborrheic keratosis, solar lentigo, squamous cell carcinoma, and vascular lesion. We used this data set to conduct our study on the detection of basal cell carcinoma using deep learning.

To collect images, the ISIC CLI tool was used. This tool allows for easy querying and bulk downloading of data from the ISIC archives. At the time of this writing, 3,408 signifies the number of basal cell carcinoma images available on the archives.

1.2.2 Detection of Image Quality

The ability to automatically detect undesirable images would enable many useful applications. Search engines would be able to automatically discard those images that are of poor quality; digital cameras and phone camera software would be able to alert the user of a poor quality shot so that they may correct the mistake; autonomous driving technology would be able to ignore poorly shot frames to reduce the chances of a mistake. To that end, the viability of using deep learning and CNNs in particular to classify images into six categories of bad lighting, Gaussian blur, motion blur, JPEG 2000, white-noise, and high quality reference images, was studied. To accomplish this task, a new dataset of images has been constructed from newly taken images, and datasets from two previous studies. The first of these datasets was used to asses the aesthetic quality of photos using machine learning techniques [4]. The second dataset was used by [5] to classify images based on different qualitative criteria. Not all images from these two datasets are used in this study since some of the images did not lend themselves to this particular task. In addition to these two datasets, additional images which were captured using a Sony Alpha 6000 camera which had been manually defocused to produce a Gaussian blur effect were included. Furthermore, the reference images were modified using software to produce a motion blur effect. Unlike previous works, we produce a general approach capable of determining the overall quality of images. The proposed method was shown to achieve a high accuracy of 81.6%.

1.2.3 Detection of Aesthetic Quality

In this research a novel approach to the problem of aesthetic ranking of images is introduced. Such a system could be used to quickly judge if a picture that has just been taken is aesthetically pleasing and would afford the photographer the opportunity to make adjustments and retake the shot. It could also be used to determine which of the thousands of snapshots taken during a weekend are worthy of being shared with

the world.

In e-commerce, computer vision and machine learning techniques have been used to find merchandise similar to previously purchased items from a buyer's profile. However, these algorithms are limited to finding similarity between various products in a catalog, or finding purchasing patterns amongst a group of buyers. This lacks the ability to account for the aesthetic characteristics that a particular buyer finds appealing. A system that is capable of building an aesthetic profile for a buyer, could be used to quickly find artwork, clothing, furniture, and other merchandise which is in agreement with the buyer's aesthetic sense, even when the buyer has not previously shown interest in items within the same category.

1.3 OUTLINE

The remainder of this dissertation is organized as follows. Chapter 2 covers the preliminary concepts used through this dissertation. Chapter 3 provides a literature review of the current state of art in the field. Chapters 4, 6, 5, propose methods for solving various image quality and aesthetic challenges. Finally, chapter 7 provides conclusions and closing remarks.

CHAPTER 2

PRELIMINARY TECHNICAL MATERIALS

The research presented in this dissertation is based on machine learning and CNNs. In this chapter the preliminary technical concepts used throughout the dissertation are covered.

2.1 MACHINE LEARNING

ML refers to a class of programs and techniques which allow a computer program to learn the behavior of a complex system without being explicitly programmed to understand it. ML includes Decision Trees, Random Forests, Support Vector Machines (SVMs), and ANNs.

2.2 CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (CNNs) are deep artificial neural networks (ANNs) applied primarily to classify images, cluster images by similarity, and perform object recognition within scenes. CNN consists of convolutional and sub-sampling layers followed by one or more fully connected layers. The architecture of CNN is designed to take advantage of the 2D structures of an input images. In addition, compared to fully connected networks, CNNs are easier to train and have fewer parameters. To train and test CNN model, each input image will pass through a series of convolution layers and pooling for feature learning. Finally, an activation function such as Softmax, Sigmoid, or ReLU is applied to classify an object. Along with other advanced machine learning algorithms, CNNs have become fundamental to the field of computer vision.

2.2.1 Convolution Operation

The key layer in any CNN, is the convolution layer. This layer performs a mathematical convolution operation, denoted by $*$. Equation 2.1 shows the convolution operation.

$$O[u,v] = F[m,n] * I[u,v] = \sum_{m}\sum_{n} F[m,n] \cdot I[u+m, v+n] \tag{2.1}$$

2.2.2 Filtering

Filters are nothing more than a mask that is applied to a part of an image to determine if a segment of an image contains a particular feature such as a diagonal line, or a horizontal line. The filter is then moved throughout parts of the image by moving it from its current location to the next location by an offset known as a stride (rate of movement). The resulting matrix is known as a feature map and is always smaller than the starting image.

Output Size As convolutions are performed, the resulting feature map will have a size that is different from the input. It is important to be able to calculate the resulting feature map size when designing a network to ensure optimal performance. We can calculate the output width and output height using equations 2.2 and 2.3, respectively.

$$width_{output} = \frac{W - F_{\mathrm{w}}}{S_{\mathrm{w}}} + 1 \tag{2.2}$$

$$height_{output} = \frac{H - F_{\mathrm{h}}}{S_{\mathrm{h}}} + 1 \tag{2.3}$$

where W is the width of image, H is the height of image, F_{w} is the width of the filter, F_{h} is the height of the filter, S_{w} is the horizontal stride, and S_{h} is the vertical stride.

Pooling

A common practice in the design of CNNs is to periodically, or often after every convolutional layer, insert a pooling layer. The function of the pooling layer is to reduce the size of the feature maps that are being worked with in order to lower the number of parameters which must be learned and hence increase the training performance of the network. There are different types of pooling algorithms, however they all work on the basic idea of having some size $N \times M$ which is used to combine values. For example, given a 4×4 starting image and a pooling layer of 2×2, the values of [0,0], [0,1], [1,0], and [1,1] will be combined to become the new [0,0] cell. [2,0], [2,1], [3,0], and [3,1] will be combined to become the new [1,0] cell. [0,2], [0,3], [1,2], and [1,3] will be combined to become the new [0,1] cell. [2,2], [2,3], [3,2], and [3,3] will be combined to become the new [1,1] cell. The resulting image will be a 2×2 image.

The pooling of values is often performed using a max function, where the maximum value of the cells being combined is taken as the new value. Other functions such as min, or average can also be used to perform the pooling function.

Image Resizing

In order to make training easier and to allow for consistent network designs, the input images to CNNs are often resized to something which is lower than the actual image size. Images are often reshaped to a square size so that square filters can be applied to them with ease.

2.2.3 Optimizers

During training the actual labels of the data are compared to the predicted label and the cost function is minimized. A cost function of zero denotes completed learning. An optimizer is needed to minimize the cost function. TensorFlow provides a variety of

optimizers which can be used during the training process. The three most commonly used optimizers are Adam, RMSProp, and SGD.

SGD The Stochastic Gradient Descent (SGD) optimizer allows for updating of network weights per training image. Because of the noisiness of per image weight updates, a special case of SGD is implemented TensorFlow known as Mini-Batch Gradient Descent. This special case of SGD is where the number of samples is more than one. SGD is implemented by TensorFlow and uses equations 2.4 and 2.5 for weight updates.

$$w_t + 1 = w_t - \eta \frac{\partial C}{\partial w_t} \tag{2.4}$$

$$\frac{\partial C}{\partial w_t} = \nabla_w C(w_t; x^{(i+i+n)}; y^{(i:i+n)}) \tag{2.5}$$

where w_t are the weights at step t, η is the learning rate, n is the number of data points in the batch, and $C()$ is the cost function. $\nabla_w C(w_t)$ is the gradient of the weight parameters for an image x, and its corresponding label y.

One of the disadvantages of SGD is oscillations during the updating of weights. These oscillations often interfere with the learning process and increase the time to convergence. To overcome this, a technique known as momentum based gradient descent is employed.

$$V_t = \lambda V_{t-1} + \eta \frac{\partial C}{\partial w_t} \tag{2.6}$$

$$\frac{\partial C}{\partial w_t} = \nabla_w C(w_t; x^{(i+i+n)}; y^{(i:i+n)}) \tag{2.7}$$

$$w_{t+1} = w_t - V_t \tag{2.8}$$

Momentum based gradient descent uses equations 2.6, 2.7, and 2.8 during weight update, where V is the velocity initialized to 0, λ is the momentum carried forward

from the previous update, w_t are the weights at step t, η is the learning rate, n is the number of data points in the batch, and $C()$ is the cost function. $\nabla_w C(w_t)$ is the gradient of the weight parameters for an image x, and its corresponding label y [6].

Adam Optimizer

The Adam optimizer was introduced to combine the benefits of several optimization methods, including Nesterov momentum, AdaGrad and RMSProp. Weight updates in Adam are handled by equation 2.9.

$$w_t^i = w_{t-1}^i - \frac{\eta}{\sqrt{\hat{v}} + \epsilon} \cdot \hat{m}_t \tag{2.9}$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{2.10}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{2.11}$$

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) G \tag{2.12}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)[G]^2 \tag{2.13}$$

$$G = \nabla_w C(W_t) \tag{2.14}$$

where w_t are the weights at step t, η is the learning rate, n is the number of data points in the batch, and $C()$ is the cost function. $\nabla_w C(w_t)$ is the gradient of the weight parameters for an image x, and its corresponding label y. β_i is used to determine the information passed forward from previous layers and m_t is the running average of the squared gradients.

2.3 RESNET

Since the introduction of AlexNet in 2012, researchers have begun to increase the depth of their networks with reasonable success. However, the depth wise growth reached a limit with deeper networks leading to higher training accuracy. Table 2.1 shows the error rates of various networks as presented by He et al. [7]. This clearly shows the performance degradation with the 34 layer plain network showing worse performance than the much shallower VGG-16 network. The interesting thing noted by the authors was that this degradation is not attributed to overtraining, and they noted that this issue was not caused by the vanishing gradient problem since batch normalization was used and neither forward nor backward signals vanish. Although some research has been carried out to understand the mechanism by which Residual Netoworks (ResNets) improve accuracy, this topic is not yet well understood and more research is needed [8, 9].

The ResNet architecture has four main concepts: 1) the majority of convolutional layers have 3×3 filters and have the same number of filters for the same output map size; 2) the number of filters is doubled when the feature map size is doubled which preserves the time complexity per layer; 3) downsampling is applied directly by convolutional layers with a stride of 2; 4) shortcut connections are made from the input of each layer to the following layer's output, prior to activation [7]. Figure 2.1 shows the fundamental residual block. The proposed model of this paper is inspired by these concepts.

2.4 INTERRATER AGREEMENT

Interrater agreement is the measure of closeness between ratings provided by raters for the same subject. There have been various methods and models proposed to address this challenge, however most of these methods can only be used on a single target.

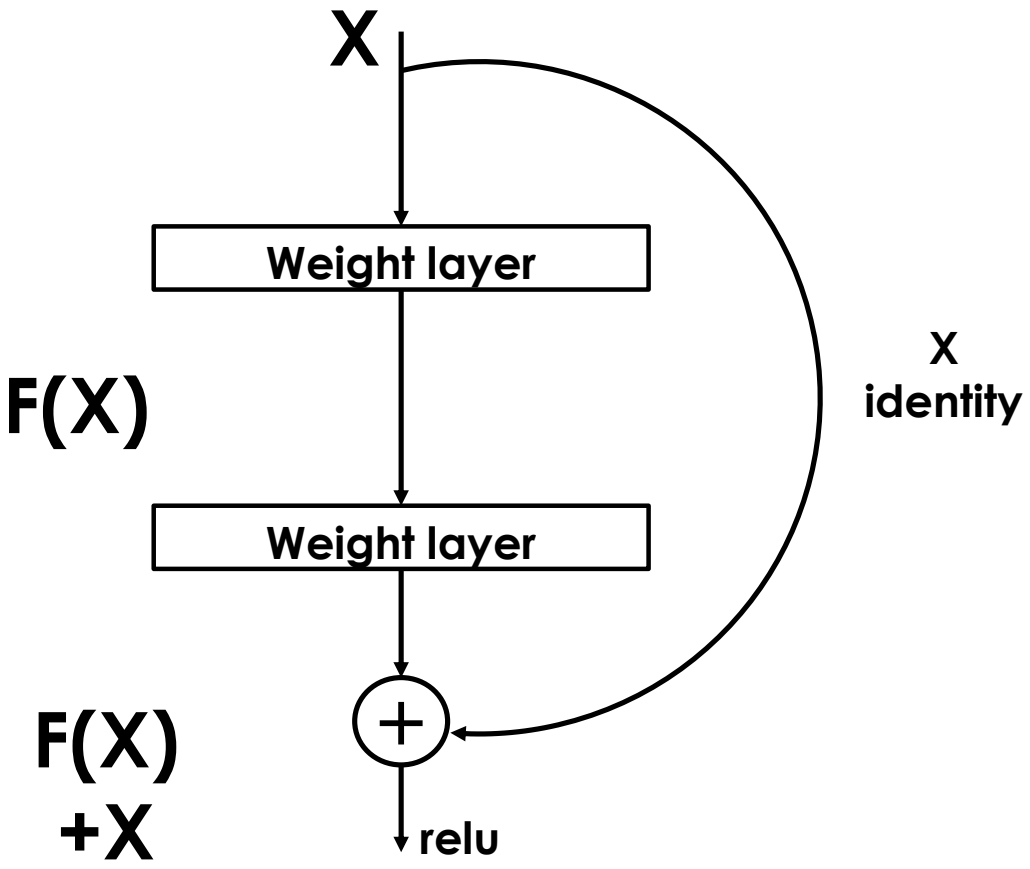


Figure 2.1: The fundamental residual block showing the shortcut connection and activation layers [7]

A more systematic approach is required to judge general agreement between the model's ratings and those provided by human raters. Burke, Finkelstein, and Dusig [13] proposed the use of average deviation indices as a means to estimate interrater agreement.

$$AD_{M(j)} = \frac{\sum_{n=1}^{N} |x_{jk} - \bar{x}_j|}{N} \tag{2.15}$$

where $AD_{M(j)}$, Equation 2.15, is the average absolute difference from the mean for a particular subject calculated for item j. N is the total number of judges, $\bar{x}_j$ is the arithmetic mean for all scores provided by all judges for item j, and x_{jk} is the score for item j by judge k.

The index across all items is then calculated by $AD_{M(J)}$ as defined in Equation 2.16:

$$AD_{M(J)} = \frac{\sum_{j=1}^{J} AD_{M(j)}}{J} \tag{2.16}$$

where $AD_{M(J)}$ is essentially the arithmetic mean of $AD_{M(j)}$ calculated for J items. In this study, an $AD_{M(J)}$ of zero indicates complete agreement, and two would indicate complete disagreement.

Table 2.1: Error rates (%, 10-crop testing) on ImageNet validation. All data was presented by He et al. [7].

model	top-1 err.	top-5 err.
VGG-16 [10]	28.07	9.33
GoogLeNet [11]	-	9.15
PReLU-net [12]	24.27	7.38
plain-34	28.54	**10.02**
ResNet-34 A	25.03	7.76
ResNet-34 B	24.52	7.46
ResNet-34 C	24.19	7.40
ResNet-50	22.85	6.71
ResNet-101	21.75	6.05
ResNet-152	21.43	5.71

CHAPTER 3
LITERATURE REVIEW

Interest in computer vision techniques has been increasing at an astonishing pace. The literature on the topics of deep learning, CNN, and computer vision is vast. In this chapter, a literature review of recent publications in regards to photo quality assessment, aesthetic ranking, and medical diagnosis classification is presented.

3.1 METHODS FOR CLASSIFICATION OF PHOTO QUALITY PARAMETERS

3.1.1 Learning Based Approaches

Darunga and Konik introduced a neural network based approach for the analysis of blurry regions within photographs in order to extract meta-data context from the images [14]. This method relies on passing segments of photographs to a multi-layer neural network or other machine learning based algorithm which can then analyze the blur present. This approach does not aim to determine whether or not an image is of high quality, it only analyzes the blur that is present in each region.

Liu, Li, and Jia [15] used a combinational method which used Local Power Specturm Slope, Gradient Histogram Span, and Maximum Saturation to first detect blurry images, then a Bayes classifier is trained to detect the type of blur by training on patches of images to make the process more efficient. Based upon similar ideas, Su, Lu, and Tan [16] and Gu et al. [17] were able to obtain improved results. While this approach is able to determine the type of blur, it is not able to determine if an image is blurry on its own.

Yang et al. [18] have proposed a method for focus quality measure using a deep learning approach for images taken from microscopes. They used a dataset of 384 in-focus Hoechst stain images of U2OS cells and used a synthetic de-focusing algorithm to produce out of focus images. This approach however does not look at the overall quality of an image, it focuses on determining if there is Gaussian blur present on particular segments of images.

Bianco et al. [19] used a CNN to produce an image quality score by average-pooling the scores predicted on multiple sub-regions of an image. The authors used a CNN that was originally trained to discriminate 1,182 visual categories fine-tuned for category-based image quality assessment tasks. The CNN is used to extract features which were then sent to an SVR to predict the quality score. The authors state that they were able to achieve a 0.91 Linear Correlation Coefficient with human subjective scores. This work does not attempt to make a categorization of images into poor quality categories.

Previous studies mostly focused on performing a binary classification for one image quality problem at a time. However, the methodology presented in this study is able to detect motion blur, Gaussian blur, poor lighting, white-noise, and JPEG-2000 compression errors in an image. To our knowledge at the time of this writing, there are no existing works which perform general classification of images based on image quality problem category.

Non-Learning Based Approaches

Non-learning based approaches for detecting blurry images have been used for many years. While these approaches produce fairly good results in detecting blurriness, they are not capable of categorizing an image into different poor image categories, and are not capable of determining if an image has other quality problems such as white noise or compression artifacts. We have however included the previous work

using these non-learning based approaches for completeness.

Many developed blurred image detection methods are based on edge sharpness information. [20] proposed a non-reference blur metric analyzing the spread of the edges in an image. [21] proposed a non-parametric image blur image based on edge analysis obtained by combining standard deviation of the edge gradient magnitude profile and the value of edge gradient magnitude using a weighted average. [22] developed a new blur detection scheme based on the edge type and sharpness analysis using Harr wavelet transform. In addition to detect the blurred images, this method was able to determine the extent of blurriness.

In 2008, [23] proposed a new measure based on computing the prediction residue of neighboring pixels in images and computing the variance to measure the blurriness without reference. [16] proposed an automatic image blurred detection and classification technique based on a new blur metric, Singular value feature, to detect the blurred region of an image. Also, they classified the type of blurred regions into defocus blur and motion blur analyzing the alpha channel information. [15] developed a blur detection methods based on image patches, making region-wise training and classification in one image efficient. This method was also able to recognize the blur types for the detected regions using several blur features modeled by image color, gradient, and spectrum information.

Recently, [24] developed a method to compute blur detection maps based on a novel High-frequency multi-scale Fusion and Sort Transform (HiFST) of gradient magnitudes.

3.2 AESTHETIC RANKING OF PHOTOGRAPHS

Most studies cast the task of photo aesthetic assessment as a classification or regression problem, where various approaches are presented on how to extract meaningful aesthetic attributes [25]. Early research focused on handcrafted features which were



Figure 3.1: JPEG 2000 Compression Artifacts

mostly based on photographic rules and what influences the aesthetic quality of an image [26, 27, 28, 29, 30]. This approach presents limitations since a universal aesthetic model may not be applicable to particular images, and implies images sharing similar content have a common aesthetic model. Other methods thus propose a solution to this problem by using generic image features such as Fisher Vector (FV) and Bag-of-Visual-Words (BOV) to learn aesthetic attributes, and predict an image's aesthetic score [31, 32, 33].

Recently, deep learning techniques have been used for various computer vision problems [34, 35, 30, 36, 37]. While previous methods are effective, CNNs applied to photo aesthetic assessments have shown more significant state-of-the-art performances than approaches which use handcrafted and generic image features[38, 39, 40, 41, 42]. For instance, Tian et al. [43] proposed that different types of images are associated with a particular aesthetic model; based on each image's content, aesthetic features

were extracted. Kong et al. [44] presented a similar approach in which their CNN model learns aesthetic attributes by sampling image pairs with similar content. Other methods apply multi-column CNNs to learn features, such as the one proposed by Lu et al. [40] in which a double-column CNN is proposed to take local and global cues of an image; style and semantic attributes were also incorporated to improve accuracy. Wang et al. [45] expanded on the proposed model in Lu et al. [40] by constructing a parallel supervised pathway of different style CNNs to exploit each aesthetic attribute. In addition, to prevent issues originated from taking images at a fixed-size, recent studies suggested methods including scaling, cropping, and padding to alter images or proposed adding spatial pooling layers into processing of images at their original size[46, 47, 41]. Multi-task learning, in which a model simultaneously learns multiple tasks [48, 49] has also been applied to image aesthetic assessments [42, 39, 50]. Kao et al. proposed a multi-task CNN to utilize semantic recognition for aiding in learning image aesthetics [39]. However, Multi-task learning may not be suitable for all datasets because it requires images to be equipped with semantic and aesthetic labels.

Tian et al. [43] propose to construct a query-dependent aesthetic model for different testing/query images. For each query image, a query-dependent training set is built and extracts the deep aesthetic features of images within the set. Then, a query-dependent aesthetic model is learned from the images. In Tian et al.'s query-dependent aesthetic assessment system, two types of features were learned from DCNNs. One of them being the CNN features extracted using the open-source deep learning framework Caffe trained on ImageNet. The other used the DCNN model trained for aesthetic feature mining to extract deep aesthetic features. The proposed query-dependent model was found to significantly and consistently outperform state-of-the-art hand crafted feature-based and universal model-based methods.

Lu et al. [51] propose a deep neural network approach which unifies feature learning

with classifier training to estimate an image's aesthetics. The proposed novel double-column convolutional neural network (DCNN) is able to judge image aesthetics while having one column take a global view of an image and the other take a local view of the image. Lu et al. also update the DCNN in order to provide a solution to the generic image aesthetic problem by proposing a network adaptation approach for content-based image aesthetics. A regularized double-column network (RDCNN) architecture is proposed to train deep networks for image aesthetics assessment using additional attributes such as styles and semantics. When tested on the AVA test set, Lu et al.'s approaches were shown to achieve state-of-the-art results, and the presented IAD dataset improved the aesthetic assessment accuracy on the AVA test set.

Kao, Wang, and Huang [52] directly train a regression model using a neural network to interpret aesthetic quality of images. The first stage of their framework is image preprocessing which involves resizing the smallest dimension of each RGB image and cropping the image. The second stage focuses on aesthetic features learning with convolutional networks. The third and final stage of the framework is training the regression model. The regression model is trained on the basis of the learned features and the labels of the images, which are the average score of user ratings for each image. Through this, the model is able to automatically predict the aesthetic score for testing images. The network is trained on the AVA dataset and the experimental results display that the aesthetic features learned by the convolutional network perform better than existing features. The results also showed that the regression model can automatically assess an image's aesthetic quality and that Kao et al. 's method outperforms the state-of-the-art methods.

Jin et al. [53] present the inception module into image classification. They build a novel Deep convolutional neural network, codenamed ILGNet (I: Inception, L: Local, G: Global) using multiple inception modules for image aesthetics assessment. The

Inception network is 13 layers deep when counting only layers with parameters and includes a connection between the layers of local features to the layer of global features. The output layer gives the classification result of either low or high aesthetic quality of an image. The ILGNet was first trained on the ImageNet and the, Jin et al. fixed the inception layers and fine tuned the connected layer that contains global and local features on the AVA dataset. The proposed ILGNet was tested among other state-of-the-art methods on the AVA dataset and was found to outperform these methods and go deeper than current DCNN used for image aesthetic quality assessments.

Dong and Tian [38] improve upon photo quality assessment by designing new aesthetic features from two different views: content-based features and rule-based features. Some of the rule-based aesthetic features include color, sharpness, and depth of field. The content-based features are based on a DCNN model that is carefully trained on the ILSVRC-2012 dataset to predict photo quality using the DCNN descriptor. Dong and Tian then combine these features using the Multi-Kernel Learning (MKL) method to predict multi-level photo quality. This is done by choosing different kernels for different aesthetic features to create a group of kernel matrices which will ultimately create a classification model. The proposed rule-based features and content-based feature was compared with state-of-the-art methods for binary photo quality prediction on the datasets, CUHKPQ and AVA. The results showed that the DCNN descriptor outperformed all other features on both datasets, and that the rule-based features also had high efficiency. The MKL method, as well, displayed promising performance and demonstrated the importance of combining aesthetic features from different views.

Dong et al. [47] implement and adopt a deep convolutional neural network that has the same architecture as a previous study (Krizhevsky et al., 2012) which was used for image classification. Dong et al., however, use the neural network to understand images in order to perform a photo aesthetic quality assessment. This is done by

training the neural network on the ILSVRC-2012 training set which extracts images from the ImageNet database. For each image, the last hidden layer of the convolutional neural network produces dimension activations to be DCNN_Aesth features. A classifier is also trained to predict the aesthetic quality of new images. To avoid the loss of information when scaling an image to a fixed size, Dong et al. adopted the idea of spatial pyramid on the images, producing DCNN_Aesth_SP features. The effectiveness of deep convolutional neural networks for photo quality assessment was tested on the datasets CUHKPQ and AVA, along with other state-of-the-art methods. On both datasets, DCNN_Aesth_SP features got the best performance, followed by DCNN_Aesth features, each having high accuracy compared to state-of-the-art methods which were mostly based on hand-crafted features.

Kong et al. [44] propose to train a model through a Siamese network that takes a pair of images as input and directly predicts the relative aesthetic ranking, and the overall aesthetic scores. The CNN model unifies aesthetic attributes and photo content by sampling image pairs with similar content to learn the specific relations of attributes and aesthetics for different sub-categories. Kong et al. also present a new dataset called the "Aesthetics with Attributes Database" (AADB) in which each image is associated with a detailed score distributions, attributes annotation, and anonymized rater identities. The proposed model was tested for rating image aesthetics by comparing against many baselines, analyzing the dependence of model performance on the model parameters and structure, and comparing the model's performance with human aesthetic rankings. The results indicated that the model is efficient on existing classification benchmarks, achieves state-of-the-art classification performance on the AVA benchmark, performs as well as the average human rater, but lags behind more consistent people who label large batches of images.

Mai, Jin, and Liu [41] propose a deep Multi-Net Adaptive Spatial Pooling Convolutional Neural Network (MNA-CNN) architecture which can directly learn aesthetic

features from the input images at its original size and aspect ratio. The method serves as a solution to the problem of preserving the quality of images when performing photo aesthetics assessment. The deep network architecture is composed of several sub-networks, each having an adaptive spatial pooling layer with a different pooling size. A scene-aware aggregation layer is also built to enhance the MNA-CNN to combine the predictions from the several sub-networks. Mai et al. use the VGG network (VGG-Net) pre-trained on the Imagenet dataset as the base network architecture for supervised feature transfer and the Places205-GoogLeNet for the scene-categorization ConvNet component. The method was tested on the AVA benchmark by assigning binary aesthetics labels to each image and the results showed that padding an image performs better than cropping, but worse than scaling an image. The overall performance of the MNA-CNN demonstrates that the method can significantly improve the state-of-the art results in photo aesthetics assessment.

Deng, Loy, and Tang [25] present and review various attempts of image aesthetic assessment that differentiate between high-quality and low-quality photos. One of the reviewed conventional methods for image quality assessment is based on handcrafted features which include simple-image, general-purpose, and task-specific features. The other approach reviewed is based on deep learning which includes generic deep features and learned aesthetic deep features with CNNs. The results of the methods discussed throughout the paper indicate that those that involve deep learning based approaches are more effective in understanding an image's aesthetic quality, when tested on images from the AVA dataset. The results also showed that a better performance was achieved with a 2-column CNN baseline than compared to a 1-column CNN. Deng et al. also explore the approach of automatic image cropping by adapting the learned aesthetic-classification CNN to complete aesthetic-based image cropping.

3.3 MEDICAL DIAGNOSIS CLASSIFICATION

Numerous studies have been conducted surrounding the topic of assessing focus quality in images. In [54], Lopez et al. proposed an algorithm that identifies blurred regions in images due to poor focusing. During training, the algorithm extracted Haralick features from 48,000 tiles with the size of 200x200 pixels. Each test tile was classified as in-focus or blurred by utilizing the extracted features and a Decision Tree. However, their method poses limitations because it can only be applied to images with a high level of blurriness. Jiminez et al. presented a similar approach that used Otsu thresholding to extract the tissue map [55]. Once the tissue was divided into pixel tiles, the Tenegrad statistics, Cumulative Probability of Blur Detection contrast, and entropy were calculated for all tiles. These calculations were used to determine whether a tile was considered out-of-focus.

In recent years, deep learning has enabled there to be vast improvements in various research fields, particularly computer vision [36, 30, 35, 56]. Neural networks are able to produce state-of-the-art performances when classifying images despite having no human intervention. Due to their large success, neural networks are increasingly being applied to the task of identifying blurry and out-of-focus regions in medical imaging [57, 58, 59, 60, 18, 61].For instance, Yang et al. [18] developed a deep neural network that automatically identified the absolute focus quality of a single image in isolation. The network was trained on microscopic images of U2OS cancer cells and was used to identify the extent of the image blur and if this image blur was well-defined. When tested, the model showed it was more accurate in classifying out-of-focus microscope images than previous methods, and that the network could be generalized to different images and cell types. Within the rise of deep learning in focus quality assessments, CNN have become more prevalent as well [57, 58, 59, 62, 60, 18]. In [62], a CNN-based system called DeepFocus is proposed to automatically detect blurry regions in histopathological images. The researchers trained the network on H&E and IHC-

stained slides and tested it on digital images that contain patients' various diseases. They found that their method had a higher accuracy than that of [54]. Campanella et al. implemented a deep ResNet model and applied sharpness metrics for blur detection in images of different tissues and image sizes [63]. The prediction accuracy of this model was compared to that of a random forest framework, with the ResNet slightly outperforming the random forest. Tiba et al. constructed an ensemble-based outlier detection method, which involves the application of convolutional neural networks and a SVM classifier [60]. In their proposed method, the hybrid CNN-SVM model, the CNNs were trained on retinal and skin lesion images, some of which were of low quality. The CNN-SVM was shown to outperform classic CNNs in filtering images with anomalies.

Other works have also applied neural networks to deblur medical imaging and to improve its low quality [57, 58]. Jiang et al. implemented a blind convolution framework to deblur a single microscopic image on defocus or motion blur [58]. Their method, termed deep blind microscopic image deblurring (DBMID), first utilized a classification model to determine what the inputted image's blur type is, and then applied an appropriate deblurring model based on the image's blue type. When tested, the classification accuracy of blur type was found to be 99.77% and other experimental results showed that DBMID can remove the defocus blur and extend the depth of field of the imaging system. In [57], Ali et al. proposed a multi-scale and single-stage convolutional neural network detector to identify artifacts that impede the visual interpretation of endoscopy videos. They also introduced artifact type-specific restoration by using Generative Adversarial Networks (GANs). While these methods have been effective, it does take a long time in order for the images to be deblurred.

CHAPTER 4

PHOTO QUALITY CLASSIFICATION USING DEEP LEARNING

The detection of poor quality images for reasons such as focus, lighting, compression, and encoding is of great importance in the field of computer vision. The ability to quickly and automatically classify an image as poor quality creates opportunities for a multitude of applications such as digital cameras, phones, self-driving cars, and web search technologies. In this chapter an end-to-end approach using CNNs is presented to classify images into six categories of bad lighting, Gaussian blur, motion blur, JPEG 2000, white-noise, and high quality reference images. Finally, the application of the developed model was evaluated using images from the German Traffic Sign Recognition Benchmark. The results show that the trained CNN can detect and correctly classify images into the aforementioned categories with high accuracy and the model can be easily re-calibrated for other applications with only a small sample of training images.

4.1 INTRODUCTION

The classification of unwanted images are of great importance in the field of computer vision. The ability to automatically detect undesirable images would enable many useful applications. Search engines would be able to automatically discard those images that are of poor quality; digital cameras and phone camera software would be able to alert the user of a poor quality shot so that they may correct the mistake; autonomous driving technology would be able to ignore poorly shot frames to reduce the chances of a mistake. The types of problems that may exist within an image are

from a wide range of issues related to improper photography techniques, such as bad lighting, or out of focus images, to encoding issues which causes unwanted artifacts such as what occurs in a JPEG-2000 image. There have been numerous studies and methods proposed for the detection of poor quality images. However, the use of deep learning for the purposes of image quality detection is limited within the literature.

Deep learning has shown great promise in solving complex tasks by using a black-box approach to the problem. With the advent of advanced deep learning models such as Convolutional Neural Networks (CNN), researchers have been able to produce remarkable results when it comes to the field of image recognition. Handwriting detection and simple object recognition have become almost common place within the computer vision field. These are tasks that only a short time ago were thought of as "cutting edge" yet today, these tasks can be performed with ease through the use of deep learning toolkits such as Google's TensorFlow and Microsoft's CNTK and other frameworks built on top of these packages such as Keras.

In this paper, a study is performed to assess the viability of using deep learning and CNNs in particular to classify images into six categories of bad lighting, Gaussian blur, motion blur, JPEG 2000, white-noise, and high quality reference images which are subjectively considered to be of good quality. To accomplish this task, a new dataset of images has been constructed from newly taken images, and datasets from two previous studies. The first of these datasets was used to asses the aesthetic quality of photos using machine learning techniques [4]. The second dataset was used by [5] to classify images based on different qualitative criteria. Not all images from these two datasets are used in this study since some of the images did not lend themselves to this particular task. In addition to these two datasets, additional images which were captured using a Sony Alpha 6000 camera which had been manually defocused to produce a Gaussian blur effect were included. Furthermore, the reference images were modified using software to produce a motion blur effect. Unlike previous works,

we produce a general approach capable of determining the overall quality of images.

The rest of this chapter is laid out as follows. In section 4.2 preliminary materials will be presented to introduce the types of image quality issues and their causes, and a short literature review to introduce previous works in this field, as well as a brief introduction to Convolutional Neural Networks, TensorFlow and Keras. Section 4.3 will introduce the methodology of the proposed model. Section 4.4 will analyze the results of the experiments. Section 4.5 analyzes the application of the proposed model to self-driving cars. Finally, section 4.6 provides concluding remarks and future steps. The code for the experiments is provided as a Jupyter notebook which is available for download.

4.2 PRELIMINARY MATERIALS

4.2.1 Image Quality Issues

There are many reasons for which a photograph can be considered as having poor quality. In this study, we have focused on five of these categories, Bad Lighting, Gaussian Blur, JPEG-2K, White Noise, and Motion Blur.

Bad Lighting refers to images which have been shot without adequate light for the timing and aperture of the camera. This can cause images to look dull or dark.

Gaussian Blur is caused by an out of focus camera. This type of blur could be caused by faulty auto focus mechanism on a camera, a lens which is poorly constructed, or an image focused on the wrong subject [64]. [65] describe the types of image quality and blur problems that may exist within an image. Figure 4.1 shows a sampling of images with Gaussian Blur.

JPEG-2K The JPEG-2K image format was introduced in the year 2000. While there are many advantages to this image compression format, one disadvantage is

that the JPEG-2K format is much less content adaptive than the older JPEG format meaning that the image quality can be vastly different given the same bit-rate for different content. This makes it likely to end up with compression artifacts within the image.

White Noise is the exhibition of random white grains throughout an image. This noise is often caused by film grain, various sensors and circuits such as CCDs in digital cameras and detectors in a scanner, or it could be caused by the communications channel or signal quantization [66].

Motion Blur is caused by taking an image were the subject is not stationary in relation to the camera. This can be cause by either a shaking or moving camera, or it could be cause by a subject which is moving.



Figure 4.1: Sample Gaussian Blur Images

4.3 METHODOLOGY

4.3.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are deep artificial neural networks (ANNs) applied primarily to classify images, cluster images by similarity, and perform object recognition within scenes. CNN consists of convolutional and sub-sampling layers followed by one or more fully connected layers. The architecture of CNN is designed to take advantage of the 2D structures of an input images. In addition, compared to fully connected networks, CNNs are easier to train and have fewer parameters. To train and test CNN model, each input image will pass through a series of convolution layers and pooling for feature learning. Finally, an activation function such as Softmax, Sigmoid, or ReLU is applied to classify an object. Along with other advanced machine learning algorithms, CNNs have become fundamental to the field of computer vision. In our approach, we build and train a CNN with 3 convolutional layers to classify images into the six predefined categories. We show that our approach is capable of achieving relatively high accuracy with just a limited dataset used for training, and furthermore, we analyze and show that for all six classes, we can achieve high specificity and relatively high sensitivity.

Filtering

Filters are nothing more than a mask that is applied to a part of an image to determine if a segment of an image contains a particular feature such as a diagonal line, or a horizontal line. The filter is then moved throughout parts of the image by moving it from its current location to the next location by an offset known as a stride (rate of movement). The resulting matrix is known as a feature map and is always smaller than the starting image.

Output Size Let W be the width of image, H the height of image, F_{w} the width of the filter, F_{h} the height of the filter, S_{w} the horizontal stride and S_{h} the vertical stride. Then we can calculate the output width and output height as a function of those variables.

Pooling

A common practice in the design of CNNs is to periodically, or often after every convolutional layer, insert a pooling layer. The function of the pooling layer is to reduce the size of the feature maps that are being worked with in order to lower the number of parameters which must be learned and hence increase the training performance of the network. There are different types of pooling algorithms, however they all work on the basic idea of having some size $N \times M$ which is used to combine values. For example, given a 4×4 starting image and a pooling layer of 2×2, the values of [0,0], [0,1], [1,0], and [1,1] will be combined to become the new [0,0] cell. [2,0], [2,1], [3,0], and [3,1] will be combined to become the new [1,0] cell. [0,2], [0,3], [1,2], and [1,3] will be combined to become the new [0,1] cell. [2,2], [2,3], [3,2], and [3,3] will be combined to become the new [1,1] cell. The resulting image will be a 2×2 image.

The pooling of values is often performed using a max function, where the maximum value of the cells being combined is taken as the new value. Other functions such as min, or average can also be used to perform the pooling function.

Image Resizing

In order to make training easier and to allow for consistent network designs, the input images to CNNs are often resized to something which is lower than the actual image size. Images are often reshaped to a square size so that square filters can be applied to them with ease.

4.3.2 TensorFlow

Tensor flow is an open source package created using Python which is intended for use with deep learning. The package is able to be used not only for regular fully connected neural networks but also has extensions which allow it to be used for building complex networks such as CNNs and Recurrent Neural Networks. In order to increase the training performance of large networks, TensorFlow also has a variant which is able to use the massive multi-core capability of modern GPUs to accelerate the training process. We used an NVIDIA Quadro K2200 GPU for faster training.



Figure 4.2: Dark/Bad Lighting

4.3.3 Our Convolutional Neural Network

The network consists of three convolutional layers with a Softmax activation function followed by a fully connected layer with a sigmoid activation function and finally there is a 6 node output layer with a sigmoid activation function which outputs the 6 desired

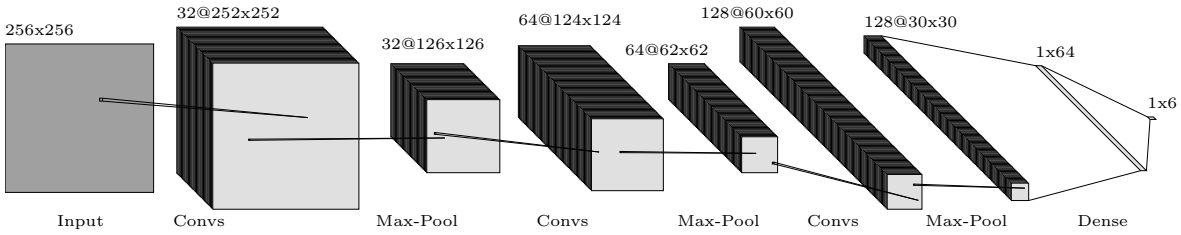


Figure 4.3: Convolutional Neural Network Architecture

classes. The first convolutional layer accepts an array of 256×256 images and applies 32, 5×5 filters to produce 32, 252×252 feature maps. We start with the larger filter size since it has been shown by Ahmed et al. [67] that a larger filter size in the upper layers produces the best results for image classification tasks. The feature maps are then normalized, a Softmax activation function is applied and a 2×2 Max Pooling is performed to obtain 32, 126×126 images. The second convolutional layer applies 64, 3×3 filters to produce 64, 124×124 feature maps As done with the first convolutional layer, the feature maps are then normalized, a Softmax activation function is applied and a 2×2 Max Pooling is performed to obtain 64, 62×62 images. This is run through a 3rd and final 3×3 convolutional layer with batch normalization and max pooling which produces 128 feature maps. The 128 feature maps are then flattened into an array of 73,856 parameters. A 20% dropout is performed before these parameters are passed into a fully connected layer (Dense in TensorFlow) with 64 nodes and a Relu activation function. Another dropout is performed with a drop percentage of 20%. This aggressive dropout is intended to keep the network from over-training on the training set and allowing for it to generalize better when applying to other types of images. Finally there is an output layer with 6 nodes with a sigmoid activation function to which all 64 nodes from the previous layer are connected.

There are a total of 7,469,145 parameters in the network with 7,468,559 trainable parameters. Table 4.1 shows the layers and the parameters for each layer and Figure 4.3 shows the architecture of our CNN.

4.3.4 Our Implementation

Our program can be split into four steps. First the images are loaded and processed. Next, we split the data into three segments for training, validation, and testing. This is followed by training the network in a two step approach by using a fast learning rate for the first 50 epochs and then reducing the learning rate and using an early stopping approach to minimize the validation loss. Finally, The network is tested against the test dataset to determine the accuracy of the network. The rest of this section is dedicated to describing the implementation methodology of the aforementioned steps. All source code and images are available for download from the GitHub provided in the references of this publication [68].

Loading Data

The images for our dataset are labeled by being placed in separate folders. In order to read the data we use Python to load the images from each directory into a distinct array. We then process the images to produce an image that is 256×256 and that image is stored as a 3 channel image within a data frame. A labels array is created for the images from each directory with 0 indicated for the clear images and 1 for the out of focus images. All images and labels are concatenated together and returned as two arrays. Prior to training, the images are split into different sets and randomized to provide accurate results.

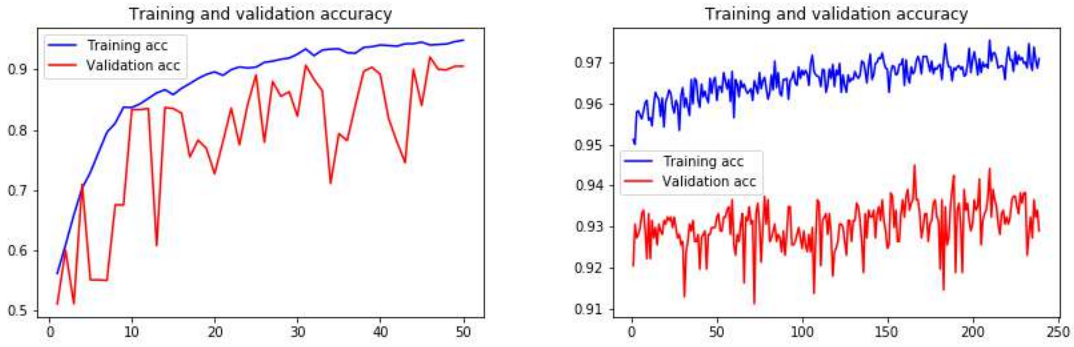
Data Splits and Training

To train the network the dataset is split into a training and a test-validation set. The test-validation set is further split into two equal parts to obtain a test set and a validation set. There are two experiments which are performed in this study with different training, test, and validation ratios.

The first experiment was performed using a ratio of 80/10/10, meaning 80% for

Table 4.1: Layers and Parameters in Proposed CNN

	Layer Type	**Output Shape**	**No. of Parameters**
1	conv2d_1 (Conv2D)	(None, 252, 252, 32)	2,432
2	batch_normalization_1	(None, 252, 252, 32)	128
3	activation_1 (Activation)	(None, 252, 252, 32)	0
4	max_pooling2d_1 (2×2)	(None, 126, 126, 32)	0
5	conv2d_2 (Conv2D)	(None, 124, 124, 64)	18,496
6	batch_normalization_2	(None, 124, 124, 64)	256
7	activation_2 (Activation)	(None, 124, 124, 64)	0
8	max_pooling2d_2 (2×2)	(None, 62, 62, 64)	0
9	conv2d_3 (Conv2D)	(None, 60, 60, 128)	73,856
10	batch_normalization_3	(None, 60, 60, 128)	512
11	activation_3 (Activation)	(None, 60, 60, 128)	0
12	max_pooling2d_3 (2×2)	(None, 30, 30, 128)	0
13	flatten_1 (Flatten)	(None, 115,200)	0
14	dropout_1 (Dropout)	(None, 115,200)	0
15	dense_1 (Dense)	(None, 64)	7,372,864
16	batch_normalization_4	(None, 64)	256
17	activation_4 (Activation)	(None, 64)	0
18	dropout_2 (Dropout)	(None, 64)	0
19	dense_2 (Dense)	(None, 6)	390
20	batch_normalization_4	(None, 6)	24
21	activation_5 (Activation)	(None, 6)	0



First 50 Epochs

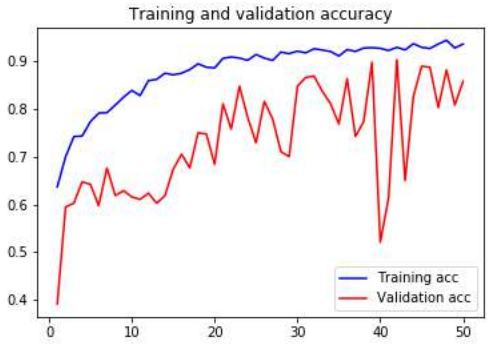
Early stopping refinement

Figure 4.4: 60/20/20 Training Accuracy History

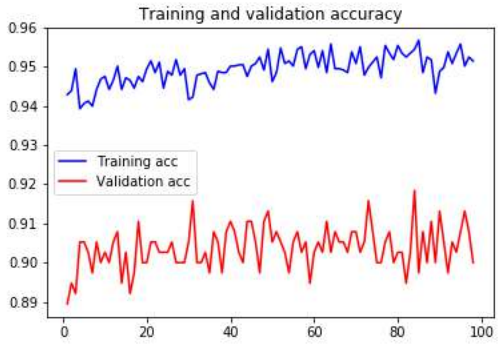
training, 10% for validation and 10% for testing. The second experiment used a 60/20/20 ratio. The network is trained for 50 epochs for the 80/10/10 set and 50 epochs for the 60/20/20 split. Figures 4.4 and 4.5 show the training accuracy history of the network after these 50 epochs. Note that during these 50 epochs, we also perform data augmentation using Keras' ImageDataGenerator which we have configured to rotate, shift, shear, zoom and flip the images. The model is then further trained using early stopping with the "patience" set to 20 which allows for further training to continue until the validation loss is not improved for 20 epochs. If that condition is reached, then training will be stopped and the previously best network will be saved. During the second training phase, the learning rate is also reduced to 0.0001 which helps in finding the most optimal solution. Figures 4.4 and 4.5 show the accuracy trend of the training and validation accuracy for the first 50 epochs and the early stopping refinement epochs.

Testing

All models were tested against the test set which was not used for the training or validation of the model. The results are analyzed by evaluating Type-I and Type-II errors and the accuracy of the predictions. To test the model, the test set was run



First 50 Epochs



Early stopping refinement

Figure 4.5: 80/10/10 Training Accuracy History

through the trained model and the predictions were converted to an array of rounded integers.

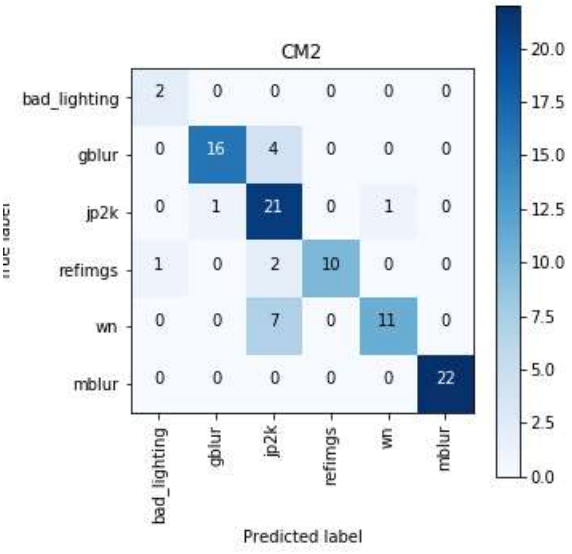


Figure 4.6: 80/10/10 Split

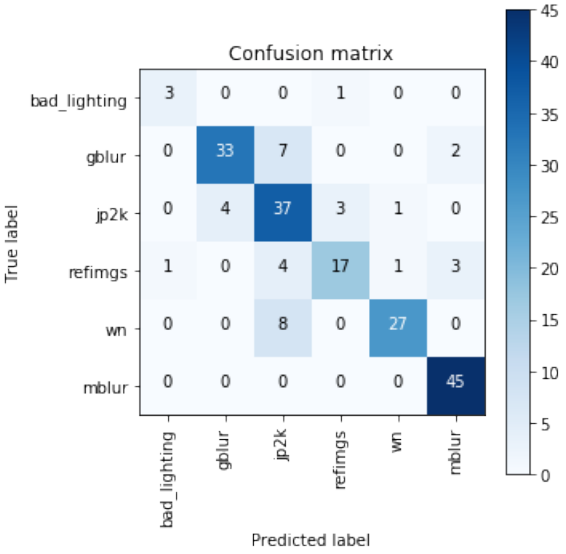


Figure 4.7: 60/20/20 Split

4.4 RESULTS

The results show a ratio split of 80/10/10 produced better accuracy than the 60/20/20 split. This suggests that with additional data a higher accuracy can be achieved. The

highest accuracy was obtained using the 80/10/10 split which reached an accuracy of 81.6%. Because many of the images in the dataset were actually in the JPEG format, there is a higher than normal rate of images which are incorrectly classified as having JPEG 2000 compression errors. This may however be correctly classified since these images may have compression artifacts which we were not able to detect. The 60/20/20 split produced an accuracy of 77%.

Figure 4.6 shows the confusion matrix for the 80/10/10 split. We can observe that the most common mistake for the algorithm is to classify images as having JPEG 2000 compression errors. As mentioned earlier this is to be expected since we are using the JPEG format for many of the images in the data set. Figure 4.7 shows the confusion matrix for the 60/20/20 experiment and like the first 80/10/10 split the results show the most common mistake is a misclassification of items as JPEG 2000.

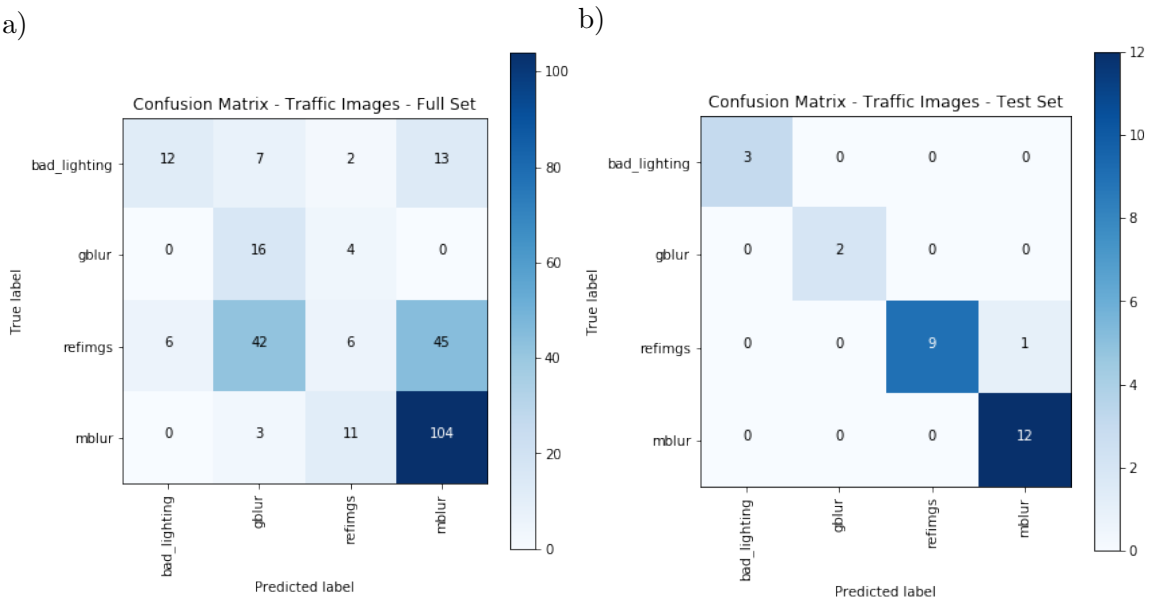


Figure 4.8: Confusion Matrix - Traffic Images

4.4.1 Additional Analysis

Sensitivity, also known as the true positive rate, measures the proportion of the positive samples which are classified as that sample. For example, of the 20 images which had been labeled with Gaussian Blur, 16 were identified as having Gaussian Blur by the network. This produces a Sensitivity of 0.8, which is saying that 80% of images with Gaussian Blur were identified as having Gaussian Blur. In other words, we can identify images with Gaussian Blur 80% of the time.

Specificity, also known as the true negative rate, measures the proportion of the samples which do not have a certain attribute as not having that attribute. For example, of the 78 samples labeled as something other than Gaussian Blur, 77 were identified as something other than Gaussian Blur and 1 was incorrectly classified as having Gaussian Blur. This produces a specificity of 0.99. This signifies that we can, with a very high accuracy, identify samples that do not have Gaussian Blur.

To get a better understanding of the performance of the classifier for each of the classes, we show the Sensitivity and Specificity for each of the six classes in Table 4.2. Sensitivity and Specificity are defined by equations 1 and 2, respectively. The results show the 80/10/10 split produces reasonably high sensitivity for all classes and very high specificity. In fact the Specificity of the classifier for the reference images category is 1.0. This suggests that the classifier can be trusted to a high degree when it classifies an image as a reference image.

$$Sensitivity = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{4.1}$$

$$Specificity = \frac{TrueNegatives}{TrueNegatives + FalsePositives} \tag{4.2}$$

Table 4.2: Classifier Performance by Class

Class	Sensitivity		Specificity	
	60/20/20	**80/10/10**	**60/20/20**	**80/10/10**
Bad lighting	0.75	1.0	0.99	0.99
Gaussian blur	0.79	0.8	0.98	0.99
JPEG-2K	0.82	0.91	0.88	0.83
Reference image	0.65	0.77	0.98	1.0
White noise	0.77	0.61	0.99	0.99
Motion blur	1.0	1.0	0.97	1.0

4.5 APPLICATION TO TRAFFIC IMAGES

Misclassification of road images is mainly caused by quality issues such as blurriness or degradation [69]. An incorrectly classified object can lead to a misinterpretation of the scenes by the autonomous vehicle. This could cause undesirable driving behavior and could place the passengers and other drivers in danger. A self-driving system equipped with a low quality image detection mechanism, could reduce the likelihood of misclassifying an object captured within the video stream by removing poor quality images. To fulfill this requirement, the model proposed in this study can be utilized to detect poor quality images used for autonomous driving systems. In order to determine how well our network performs when applied to self-driving cars, we have selected images from the German Traffic Sign Recognition Benchmark (GTSRB) [70]. The selected images with the worst lighting and worst Gaussian blur have been manually labeled and the OpenCV library on Python was used to introduce a random level of motion blur to the reference images to produce a dataset containing labeled images in four categories.

4.5.1 Initial Results

Applying the trained network, without modification, resulted in an accuracy of 50%. The majority of incorrect classifications are those reference images which were either classified as motion blur or Gaussian blur. Figure 4.8a shows the confusion matrix for the full traffic dataset. We visually observed the reference images which were incorrectly classified to determine the cause of the misclassification. Figure 4.9 shows the reference images which were classified as bad lighting, Gaussian blur, and motion blur. Upon inspection, it is noticeable that the images which were classified into one of these 3 categories are actually of poor quality and it could be argued that the majority of these images were classified correctly. This suggests that while the model may correctly have detected some degree of the attributes associated with poor

image quality, the sensitivity of our classifier to these attributes is too high; hence, to make our model useful for the purposes of reducing error rates in the traffic sign classification, we must re-calibrate our model to account for this variability. This can be accomplished utilizing a technique known as transfer learning.

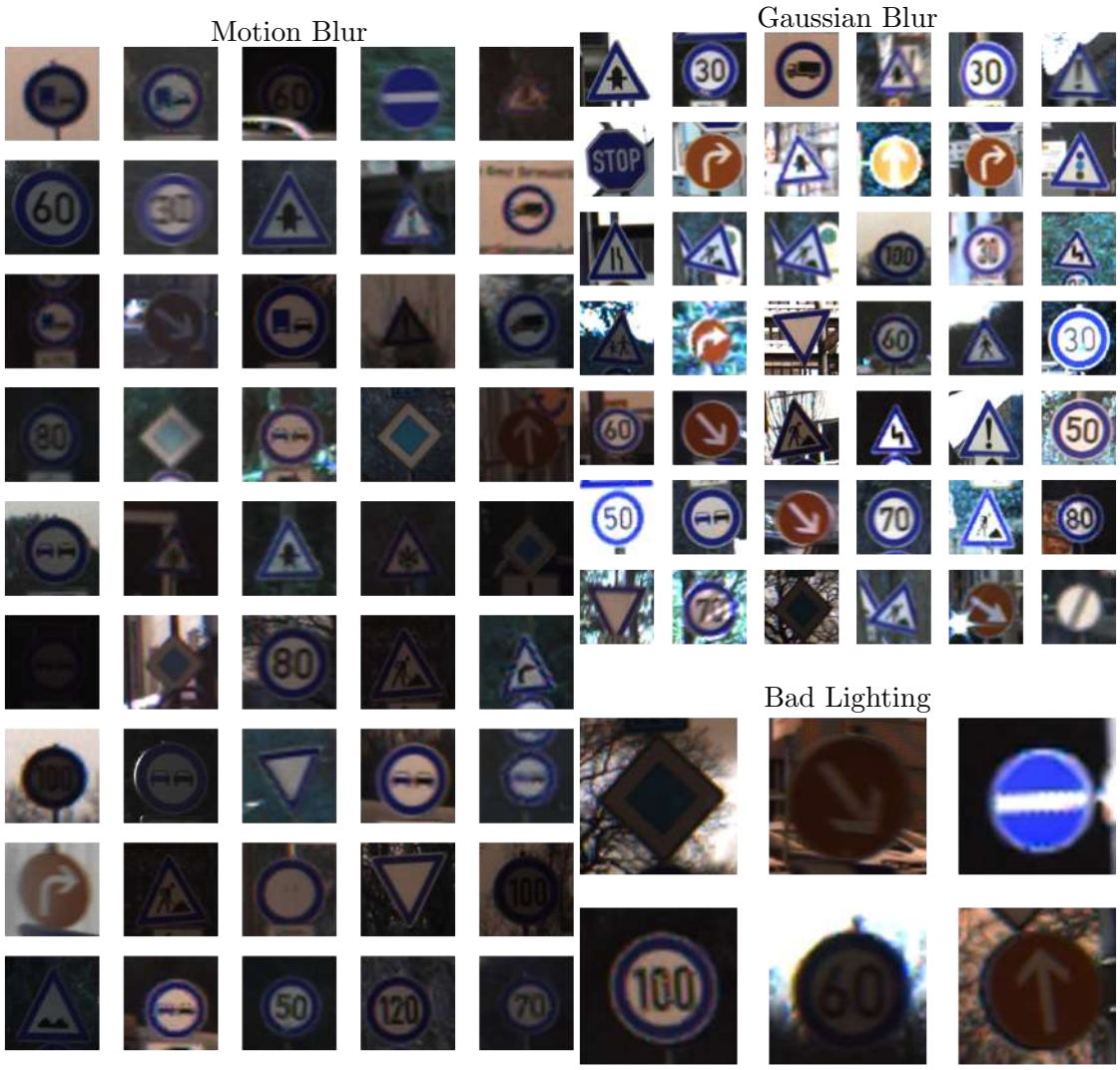


Figure 4.9: Incorrectly Classified Reference Images

4.5.2 Transfer Learning

The transfer learning process begins with loading the developed CNN model which was previously trained to classify images into the six categories of bad lighting, Gaus-

sian blur, motion blur, JPEG 2000, white-noise, and high quality reference images. According to previous studies, misclassification of road images is caused by poor quality images that are either blurry or have very poor lighting [69]. Therefor, for this task only the categories of bad lighting, Gaussian blur, motion blur and reference images are considered. First, all layers are set as not trainable. Then the last 3 layers were removed and a dense layer with 4 neurons was added. This allows for the model to be trained more quickly since only 260 parameters need to be learned. The dataset is then split into three segments of 80% for training, 10% for validation, and 10% for test. The network was trained for 50 epochs and additional training was then performed utilizing early stopping which stopped at 289 epochs. Each epoch completed within one second, and the total training task completed in 5 minutes. This retraining process accomplishes two objectives, 1) changing the number of classes from 6 to 4, and 2) allowing for the network to tune it's sensitivity for the given task. The resulting network was able to achieve a 96% accuracy on the test dataset. This shows that while the exact calibration of the model is dependent on the given task, the feature space learned through the original training process can be applied to different sets of images without loosing accuracy. The confusion matrix for the retrained network is shown in Figure 4.8b.

4.6 CONCLUSIONS AND FUTURE DIRECTIONS

With the rapid advancement of machine learning and the increasing growth of autonomous technologies, it has become even more important that machine learning algorithms be capable of detecting when the results of some action are not desirable. In this paper, results show that our proposed CNN is capable of detecting poor quality images with high accuracy. A new dataset was created to train the developed CNN model to detect images which were manually labeled into the six categories of bad lighting, Gaussian blur, motion blur, JPEG 2000, white-noise, and high quality

reference images. Finally the application of the developed model was tested to detect images which may be unsuitable to the sign classification task required by autonomous vehicles. We showed that by simply calibrating the model for a given task through transfer learning we can achieve very high accuracy for the given classification task.

This study was focused on developing a method to detect poor quality images. There are several avenues for future studies to continue upon this work. In a future study we will perform sensitivity analysis to determine how differences in camera hardware affect the accuracy of the model.

CHAPTER 5
THE EFFECTS OF IMAGE QUALITY ON DEEP LEARNING CLASSIFICATION PERFORMANCE

Medical image diagnosis is a key area of focus for the computer vision community of researchers. These techniques have advanced at a rapid rate, and are now capable of making reliable predictions of health conditions based on imaging data such as photos, X-Rays, CAT-Scans and MRI's. In this chapter, a case study on the effects of photo quality on the performance of CNN classifiers is presented. The results show that photo quality has a profound effect on classifier performance, specifically by negatively impacting sensitivity.

5.1 INTRODUCTION

One of the most common challenges faced by the global healthcare system today is providing accessible and accurate diagnoses. In fact, about 5.08% of US adults are estimated to be misdiagnosed every year [71]. Incorrect diagnoses are also prevalent among patients who have serious chronic conditions, with about 20% of these patients receiving errors in their diagnosis at the primary care level, and one in three of these misdiagnoses being harmful to patients [71]. More recently, however, machine learning and AI have shown great promise in providing accurate and efficient diagnoses [72, 73]. These emerging methods are being utilized in the medical field as diagnostic tools to aid physicians. The advancements show to transform healthcare systems by exploiting patient data to yield precise diagnoses [74, 75, 76, 77].

Despite the vast improvements made with machine learning and AI efforts, the

accuracy of patients' diagnoses remains to be compromised. This is largely due to the issue that medical images are often times out-of-focus, unevenly illuminated, noisy, blurry, or include dust artifacts [18]. Motion blur and defocus are the two most common types of imaging artifacts that can significantly degrade images [58]. Motion blur in images occurs as a result of an object's changed position during which an image is captured [78]. Images may contain defocus blur due to autofocusing systems' inadequate focusing accuracy [79]. Individual images that contain these artifacts significantly increase the risk of misdiagnosing patients and providing wrong treatment to them [80, 59]. Moreover, manually identifying out-of-focus regions is a time-consuming, subjective, and error-prone process [55].

Latest advancements in deep learning have enabled neural networks to make significant improvements in the computer vision field [81]. Due to their ability to produce near-human performances in image classification tasks, neural networks are being more commonly applied in addressing the issue of identifying blurry and out-of-focus regions in medical imaging [58, 59]. The implementation of this deep learning approach has been of great interest in recent years due to neural networks requiring no human intervention to detect discernible features. CNNs, in particular, display a great ability to learn and generalize attributes for image classification tasks [82]. Consequently, CNNs are being widely used for the classification of medical imaging in order to identify blurry images and have found state-of-the-art performance [58, 59, 60, 18].

In this work, we attempt to identify regions of blurriness and low focus quality in various medical images. We propose using a CNN to automatically identify these regions so that poor quality images are distinguished prior to being used for making medical diagnoses by AI. We anticipate that this method offers a more effective and accurate approach for making medical diagnoses, and it lowers the risk of patients receiving misdiagnoses.

5.2 METHODOLOGY

5.2.1 Data

The data for this study was download from the ISIC archives using the ISIC CLI tool. We created two datasets for this study. First, a total of 6,816 images were downloaded and processed. To prepare the images for training, each image was first resized to 244 × 244 or as close as possible with one of the dimensions remaining larger than 244. The image was then cropped on the center of the image, producing 6,816 images with consistent size. Once resized images were stored in two separate directories which signified the class of the positive or negative class of the image. The image resizing process is depicted on Figure 5.1.

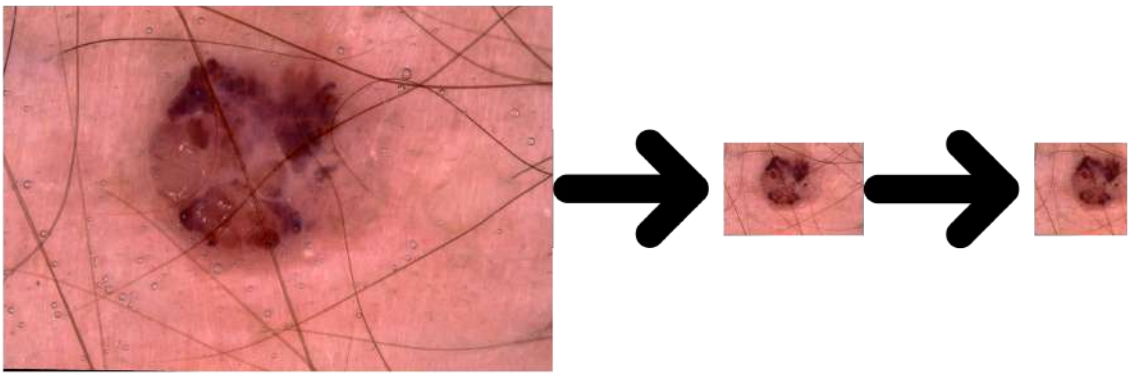


Figure 5.1: Image resizing process

Second, we created a smaller dataset focused only on differentiating between basal cell carcinoma and benign regions. The smaller dataset consisted of 1000 images in each class. We performed the same prepossessing on these images to end up with 2000 244 × 244 images.

5.2.2 Our Model

To test the hypothesis that image quality negatively impacts classification accuracy in medical imaging, we have chosen to use a model inspired by the design in chapter 6. The primary difference between the two models is the input size, output size, and minor tweaks to the filter sizes on the final convolutional layers. In all other aspects, the two networks are identical. Please refer to figure 6.3 for the network design.

5.2.3 Training

Training was performed on FAU's HPC cluster using an A100 GPU with 40 GB of video memory. To improve the generalization performance of the trained network, the Stochastic Gradient Descent (SGD) optimizer was applied in this study as it has been shown to generalize better as compared to ADAM, although ADAM has shown better training performance [83]. This is caused by the convergence behavior of SGD enabling smaller escaping time than ADAM when compared in the same basin, which in turn leads to convergence at flatter minima for SGD [84]. The initial learning rate was set to 0.1 and the momentum was set to 0.9. The batch size used for training was set to 40.

To prevent over training, early stopping was employed with a patience of ten, meaning that after 10 epochs without improvement, the training process would end and the weights from the epoch with the lowest validation loss would be reloaded. To allow for maximum training, once a validation loss plateau was reached, the learning rate would be reduce by a factor of 10 and training would continue. The minimum learning rate was $1 \times 10^{-}5$.

We trained the network using two cuts of data.

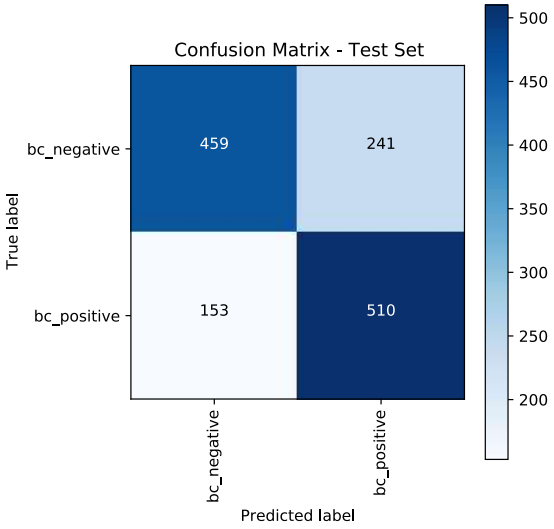


Figure 5.2: Confusion matrix depicting the performance of the classifier.

Dataset 1

The first dataset consists of 6,816 images, with 3,408 images having been diagnosed as basal cell carcinoma and the other 3,408 images belonging to one of nine groups including actinic keratosis, dermatofibroma, melanoma, nevus, pigmented benign keratosis, seborrheic keratosis, seborrheic keratosis, solar lentigo, squamous cell carcinoma, and vascular lesion.

Training was run for 21 epochs with a learning rate of 0.1, 27 epochs with a learning rate of 0.01, 16 epochs with a learning rate of 0.001, 11 epochs with a learning rate of 0.0001, and finally, 11 epochs with a learning rate of 0.00001. Weights were reloaded from the validation epoch with the lowest loss as calculated by the Mean Square Error (mse) loss function. The final training loss for the network was 0.1162 and the validation loss was 0.1801. While it is likely possible to get better validation and training results using different network architectures, the goal of this study was to determine the effect image quality issues would have on the diagnosis made by the classifier.

Dataset 2

The second dataset consisted of two thousand images, one thousand of which belonged to the benign class. The remaining one thousand images belonged to the basal cell carcinoma class.

Training was run for 23 epochs with a learning rate of 0.1, 11 epochs with a learning rate of 0.01, 11 epochs with a learning rate of 0.001, 11 epochs with a learning rate of 0.0001, and finally, 11 epochs with a learning rate of 0.00001. Weights were reloaded from the validation epoch with the lowest loss as calculated by the Mean Square Error (mse) loss function. The final training loss for the network was 0.0065 and the validation loss was 0.0235. This network showed extremely high validation accuracy of 97.25%.

5.3 ERROR INSERTION CASE STUDY

We explored the increase in error rate when images have quality problems by artificially inserting noise onto the test image dataset. Four types of noise were considered for this study; 1) Gaussian noise, this type of noise is similar to an out of focus image; 2) salt and pepper, which is a form of white noise; 3) poisson noise, another form of white noise, which is often observed on medical imaging such as MRIs and CAT scans. 4) speckle noise, caused by a scattering of the signal, is often observed on ultrasound scans.

During the testing phase, we randomly selected one of the four noise types introduced above, and injected it onto the image. It is our hypothesis that this noise would negatively impact the classifier's performance. We performed this study on both trained networks.

5.4 RESULTS

To determine the usefulness of the classifier, we will examine the specificity and sensitivity of the classifier for basal cell carcinoma.

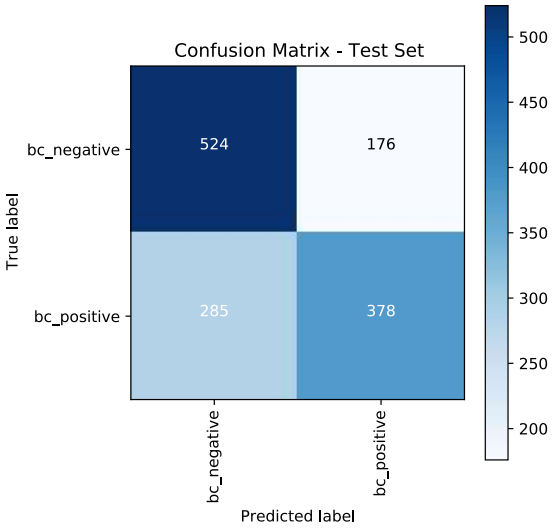


Figure 5.3: Confusion matrix depicting the performance of the classifier after the test images have been injected with noise.

5.4.1 Dataset 1

Sensitivity

Also known as the true positive rate, measures the proportion of the positive samples which are classified as that sample. Running the trained classifier against the test dataset, we obtained a Sensitivity of 0.769, meaning that 76.9% of those images which were for a confirmed basal cell carcinoma were correctly identified as such. After randomly inserting quality problems into the image, we observed that the sensitivity of the classifier had dropped to 0.570. This is a significant drop in performance and would mean that roughly, an additional 20% of those patients with basal cell carcinoma would receive a negative diagnosis which could be catastrophic since early detection is often a key indicator of long term outcome.

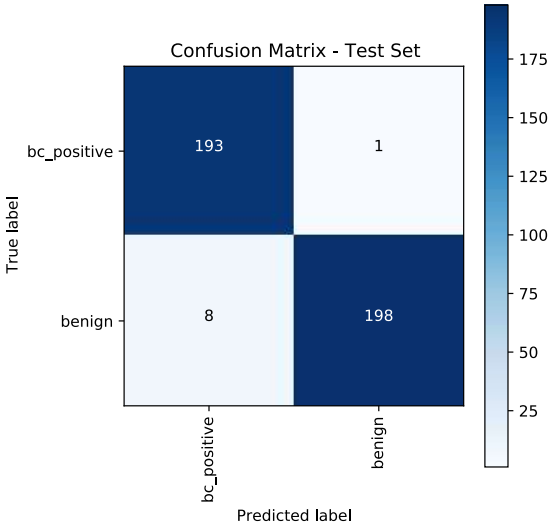


Figure 5.4: Confusion matrix depicting the performance of the classifier.

Specificity

Also known as the true negative rate, measures the proportion of the samples which do not have a certain attribute as not having that attribute. The classifier achieved a specificity of 0.656. This means that 65.6% of those who were negative, received a negative diagnosis from the classifier. After inserting errors into the images, the specificity increased to 0.749. This increase is likely attributable to the larger number of images being classified as other. While on the surface this seems like an improvement, we have to look at this in the context of the task at hand. In this case, this improvement in specificity is at the cost of sensitivity which is a more important metric in terms of detecting an illness. It is better to detect a positive case when there is nothing to worry about than to detect a negative case when the patient is in fact in need of medical attention.

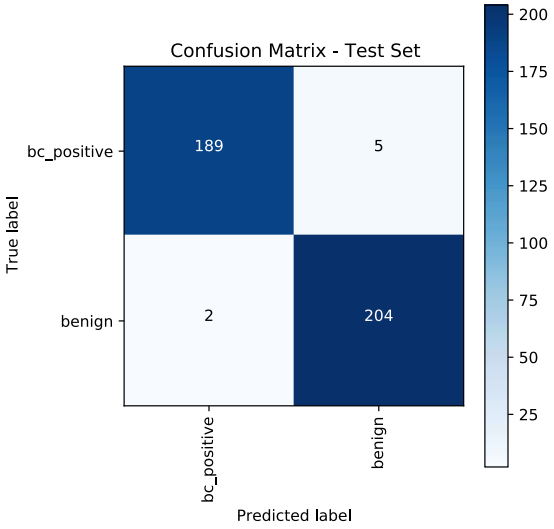


Figure 5.5: Confusion matrix depicting the performance of the classifier after the test images have been injected with noise.

5.4.2 Dataset 2

Sensitivity

Running the trained classifier against the test dataset, we obtained a Sensitivity of 0.995, meaning that 99.5% of those images which were for a confirmed basal cell carcinoma were correctly identified as such. After randomly inserting quality problems into the image, we observed that the sensitivity of the classifier had dropped to 0.974. As observed with Dataset 1, there is a drop in the sensitivity of the classifier. If we look at this from the perspective of false negative rate, the number of false negatives would be increased by a factor of five.

Specificity

The classifier achieved a specificity of 0.961. This means that 96.1% of those who were negative, received a negative diagnosis from the classifier. After inserting errors into the images, the specificity increased to 0.990. This follows the trend we have observed with Dataset 1. It should be emphasized that while having a high specificity

is desirable, it is not desirable at the expense of sensitivity which has increased by a factor of 5.

5.4.3 Conclusions

In this study, we have shown the negative effect poor quality images can have on medical image diagnosis. The negative effects are especially troubling considering that the most substantial degradation in performance was in Sensitivity which can increase the risk of missing a positive diagnosis. We showed that Sensitivity can be affected by as much as 20%.

A surprising finding is that adding noise to images can actually improve accuracy. For instance, in dataset 2, we observed that the accuracy improved from 97.5% to 98.25%. While this affected sensitivity in a negative way, in non-medical applications, this could be a viable technique for improving classification accuracy, although more research is needed to fully understand the mechanism by which this functions.

CHAPTER 6
AESTHETIC IMAGE QUALITY RANKING USING DEEP LEARNING

Aesthetic ranking of photographs is an area which has many applicable uses including social media, personal photography, and e-commerce. However, research in this area has been limited, with the vast majority of computer vision research being focused on object and scene classification. In this chapter, a unique Convolutional Neural Network design capable of learning the aesthetic qualities of images is introduced. The results show the trained model is able to outperform the baseline interrater agreement for human raters, and closely predict the human rating in more than 91% of images in the dataset.

6.1 INTRODUCTION

Ever since George Eastman made photography accessible to the average person in 1888 through his "Kodak" camera, the world has become obsessed with the art of collecting snapshots. This obsession became even more prominent with the invention of the camera phone, which put a camera in almost every person's pocket, increasing accessibility to photography, and vastly augmenting the collection of amateur photos in existence. Recently, there has been increasing interest in personal photography due to social media platforms such as Instagram, Facebook, and Flickr providing the everyday photography enthusiast with a very effective way to share photos with millions of viewers. One common feature across all of these platforms is the ability for viewers to indicate their like or dislike of a photograph, essentially a mechanism

for the crowd to react to your creation with applause, silence, or displeasure. What if it was possible for a machine to predict the crowd's reaction to a photograph? Such a system could be used to quickly judge if a picture that has just been taken is aesthetically pleasing and would afford the photographer the opportunity to make adjustments and retake the shot. It could also be used to determine which of the thousands of snapshots taken during a weekend are worthy of being shared with the world.

In e-commerce, computer vision and machine learning techniques have been used to find merchandise similar to previously purchased items from a buyer's profile. However, these algorithms are limited to finding similarity between various products in a catalog, or finding purchasing patterns amongst a group of buyers. This lacks the ability to account for the aesthetic characteristics that a particular buyer finds appealing. A system that is capable of building an aesthetic profile for a buyer, could be used to quickly find artwork, clothing, furniture, and other merchandise which is in agreement with the buyer's aesthetic sense, even when the buyer has not previously shown interest in items within the same category.

While the research community has taken an interest in this topic, automatic image aesthetics assessment remains a challenge ([29, 26, 25, 28, 34, 30, 35, 27]). This can be attributed to the subjective nature of aesthetic assessment; as the saying goes, "beauty is in the eye of the beholder". Research regarding the topic has come up with various approaches of how to address this problem. In most studies, image aesthetics assessment is considered as either a classification problem, in which an image is labeled as aesthetically good or bad, or as a regression problem, where an image is rated with a numerical aesthetic score ([52, 53]).

Many approaches have been presented in past years to explore how to extract aesthetic attributes from images[29, 26, 25, 28, 34, 30]. Initial research proposed designing aesthetic attributes focused on handcrafted features. These features are

based on photographic rules and what most likely affects people's perception of aesthetic quality of an image[38, 85]. Some handcrafted features include sharpness, rule of thirds, and colorfulness[28, 30, 26, 28]. However, the handcrafted approach poses limitations as a result of the inability of a universal aesthetic model in capturing all relevant aesthetic attributes of an image, as well as, its inability to demonstrate the impact of the set of features on overall aesthetic ranking. Due to this, other methods have proposed alternate solutions by assessing aesthetic quality through generic image features such as Fisher Vector (FV) and Bag-of-Visual-Words (BOV), which outperform methods using handcrafted features [33, 31, 32, 31, 86].

With the many usages of deep learning techniques in computer vision problems, recent research proposes using CNNs to learn effective aesthetic attributes resulting in state-of-the-art performance [34, 35, 30, 36, 37, 38, 39, 40, 41].

Although several advancements have been made in the computer vision field using CNNs, the categorization of objects and scenes has received more attention in recent studies [87, 88, 89, 90]. Hence, there have only been a limited number of attempts that apply deep learning techniques to rank or rate images based on their aesthetic qualities[38, 39, 40, 41]. While these studies display promising results, more studies are necessary to better understand how well a CNN can predict humans' aesthetic rankings of images.

In this paper, we propose a CNN to approach the problem of automatic image aesthetics assessment. We also present three different datasets consisting of images taken of forests, meadows, and bodies of water, as well as a web page where participants can manually rate the images found in the datasets. The network is trained on the three datasets to learn effective aesthetic attributes found in the images. Then, the deep network can assign an overall aesthetic rating for the set of photos based on the learned aesthetic features. The experimental results show that the proposed CNN yields an aesthetic score that displays near-human performance.

6.2 METHODOLOGY

To accomplish the task of rating images based on their aesthetic qualities, we have created a dataset of images from various photo sharing platforms such as flickr, pixabay, and unsplash. We then manually rated these images with a score from 1 to 10, with 1 being least aesthetically pleasing and 10 being the most aesthetically pleasing. These scores were then transformed to a 0 to 4 scale to reduce the number of classes and complexity of the network. To train the model, we split the dataset into three parts, 60% for training, 20% for validation, and 20% for testing. We use this dataset to train a CNN designed specifically for aesthetic ranking. Training was performed on Florida Atlantic University's High Performance Computing cluster using an NVIDIA A100 GPU with 40GBs of GPU Memory.

6.2.1 Image Dataset

The image dataset used to rate an image's aesthetic qualities was created by collecting photos from the websites flickr.com, unsplash.com, and pixabay.com. These platforms enable users to upload images to the site and provide the option to allow others to download their images. Images containing forests, bodies of water, and meadows were downloaded for this study. 2,000 images matching that criteria were downloaded at their original size.

6.2.2 Rating Images

All images in the dataset were rated based on their aesthetic features. The images were given an overall aesthetic score on a scale from 1 to 10 (where 1 is the lowest and 10 is the highest). Each image was initially rated by a number by one member of our research team and was later ranked by various participants. To collect image ratings from all participants, *PhotoRanker* has been developed in this study. PhotoRanker allows users to easily view images from the dataset and provide a rating. Each

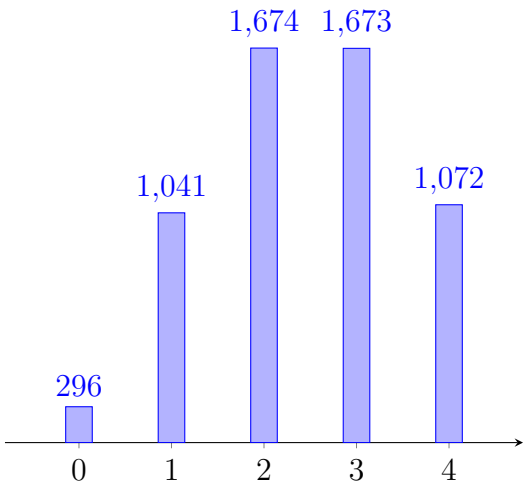


Figure 6.1: Number of times each rating from 0-4 was selected

individual ranking is collected and stored on an SQL database. For privacy reasons Personally Identifiable Information such as email addresses, IP Addresses, Location, etc. have not been collected.

Ranking Statistics

Image rankings were collected from human raters on a scale of 1-10. The rankings were modified to a scale of 0-4 to reduce the number of classes and complexity of the network. A histogram showing the number of times each rating was selected is shown in figure 6.1. After transformation, the 5755 rankings collected during the study had an average score of 2.38. To test the closeness of rankings, $AD_{M(J)}$, as defined by equation 2.16, was calculated for all images with two or more ratings. Figure 6.2 shows the number of images with $AD_{M(j)}$ obtained from human rankings, rounded to the nearest 0.1. For all images with two or more ratings, $AD_{M(J)}$ is 0.66. This was considered as the baseline for comparing the performance of our model.

As shown in figure 6.2, there is high variability in the average deviation of scores from image to image, and the distribution does not appear to be normal. For this reason, we have decided to focus on the macro results by comparing only the $AD_{M(J)}$

of the model to those received from different human raters.

6.2.3 Our Model

CNNs are well known in the world of image classification. In order to rate images based on their aesthetic parameters in a meaningful way, we have designed a network for this study to accept an image of size 1024×680 as input. The choice of input size was based on balancing the ability to train the network with minimal loss in details which may lead to decrease the aesthetic quality of images. In addition, all images were kept in the aspect ratio of 3:2 because this is how most images are experienced by viewers. The biggest hurdle to training a large network is available memory, and more specifically, memory available to the GPU. Figure 6.3 shows the architecture of the proposed network in this study. Note the lack of pooling layers which are commonly used to reduce the filter size between convolutional layers. A larger stride was applied to reduce the output volume of each convolutional layer. The use of striding instead of pooling has the advantage of a simpler architecture with less layers. [91] showed that using larger striding instead of max-pooling can reduce network complexity without any appreciable loss in accuracy. He et al. [7] which introduced ResNet uses this technique for all layers following the initial convolutional layer.

The proposed network, as shown in figure 6.3, has two main segments. The first segment in a standard ResNet design. The output space of each of the residual blocks have been reduced through the reducer phase, and the blocks were added together. Finally, the combined convolutional layers are flattened and connected to a single dense output layer with five neurons leading to a network with $+32M$ trainable parameters.

6.2.4 Training

Training was performed on FAU's HPC cluster using an A100 GPU with 40 GB of video memory. To improve the generalization performance of the trained network, the SGD optimizer was applied in this study as it has been shown to generalize better as compared to ADAM, although ADAM has shown better training performance [83]. This is caused by the convergence behavior of SGD enabling smaller escaping time than ADAM when compared in the same basin, which in turn leads to convergence at flatter minima for SGD [84]. The initial learning rate was set to 0.01 and the momentum was set to 0.01. The batch size used for training was set to 5, which was selected to reduce the memory requirements.

The learning rate is calculated for each step in the training process using the exponential learning schedule with a decay rate of 0.99. It is important to note that a step is not an epoch, instead it is the update step which can be determined from the number of steps per epoch calculated using the batch size. The learning rate can be calculated by Equation 6.1.

$$L_d = L_i * D^{\frac{s}{d_s}} \tag{6.1}$$

where L_d is the decayed learning rate, L_i is the initial learning rate, D is the decay rate, s is the current step, and d_s is the decay steps.

Training was run for 25 epochs. Weights were reloaded from the validation epoch with the lowest loss as calculated by the Mean Square Error (mse) loss function. The final training loss for the network was 0.01 and the validation loss was 0.14. Note that the performance of the model was assessed based on $AD_{M(J)}$ and the training accuracy of the network was not considered.

6.3 RESULTS

Trained network shows $AD_{M(J)} = 0.385$ for the test set, 42% lower than the $AD_{M(J)}$ of 0.66 obtained from the human ratings. This result shows that not only is the model capable of replicating human perception of aesthetic quality, it is remarkably close to replicating the personal taste of the human rater whose ratings were used for training the network. Table 6.1 shows the number of images in the test set by their human rating to model rating deviation. This shows that of the 100 images in the test set, 91 had a deviation within 1 point of the human rating. Figure 6.4 shows examples of rankings provided by the model as compared to human rankings.

Table 6.1: Number of images by deviation of model rating from human rating

$\lvert Rating_{human} - Rating_{model} \rvert$	Number of predictions
0	32
1	59
2	9
3	0
4	0

6.4 DISCUSSION AND CONCLUSIONS

With machines becoming ever more intelligent, it is inevitable that AI should find its way into activities previously thought to be reserved for human consumption. We have always thought of ourselves as special because of our ability to create science, literature, and art. In recent years, we have seen a revolution in the field of AI,

with machine learning algorithms now capable of writing articles, creating music, and even producing art [92, 93, 94]. In this research we have shown that AI can also sense aesthetics. We have introduced a unique CNN design which is capable of learning aesthetic quality of images, and predict ratings which are in high agreement ($AD_{M(J)} = 0.385$) with human provided scores, outperforming the agreement between human raters ($AD_{M(J)} = 0.66$). While we focused our effort on images of natural scenes, it would be interesting to see if these results can be replicated for image datasets that consist of portraits, artwork, architecture, and other categories.

The use cases for such a technology are numerous. For example, within personal and professional photography, an AI can inform the photographer of the aesthetics quality of the image instantaneously. This would enable the photographer to gauge if a shot is acceptable, or if it needs to be retaken from a different angle, with different lighting, or cropping. Another area where this can be useful is within social networking applications such as Instagram or online shopping marketplaces, where a user can be directed to images that are more agreeable with their own sense of aesthetics. Perhaps, a user can select a handful of images they find aesthetically pleasing, and as a result other artwork which agrees with their sense of style can be shown. We should however be cautious in the use of such technology as it could potentially cause a dwindling of art and culture by creating a feedback loop which reduces creativity by reinforcing a certain style, producing a bubble from which breaking out of can be difficult.

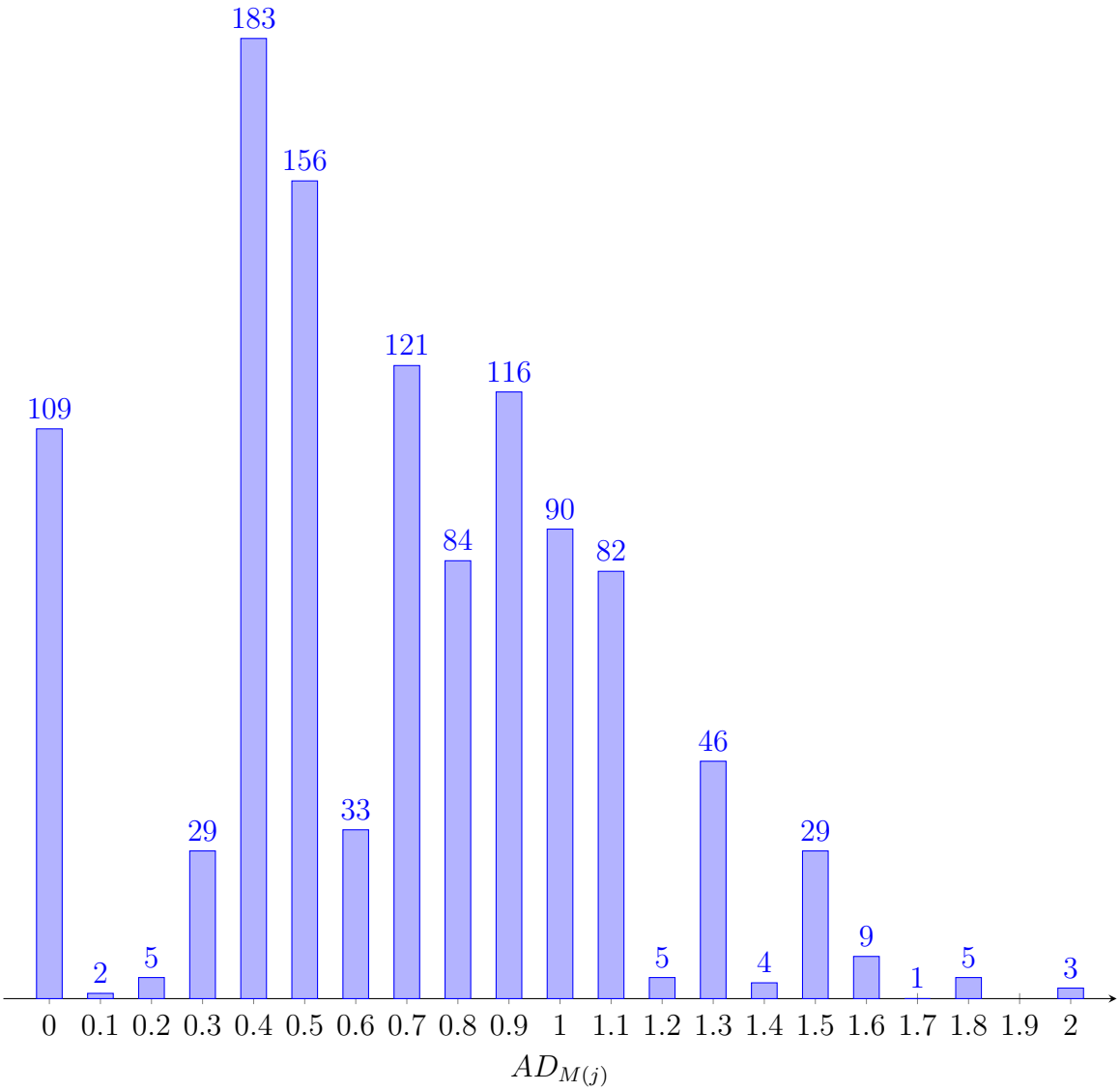


Figure 6.2: Number of images with $AD_{M(j)}$ obtained from human rankings, rounded to the nearest 0.1

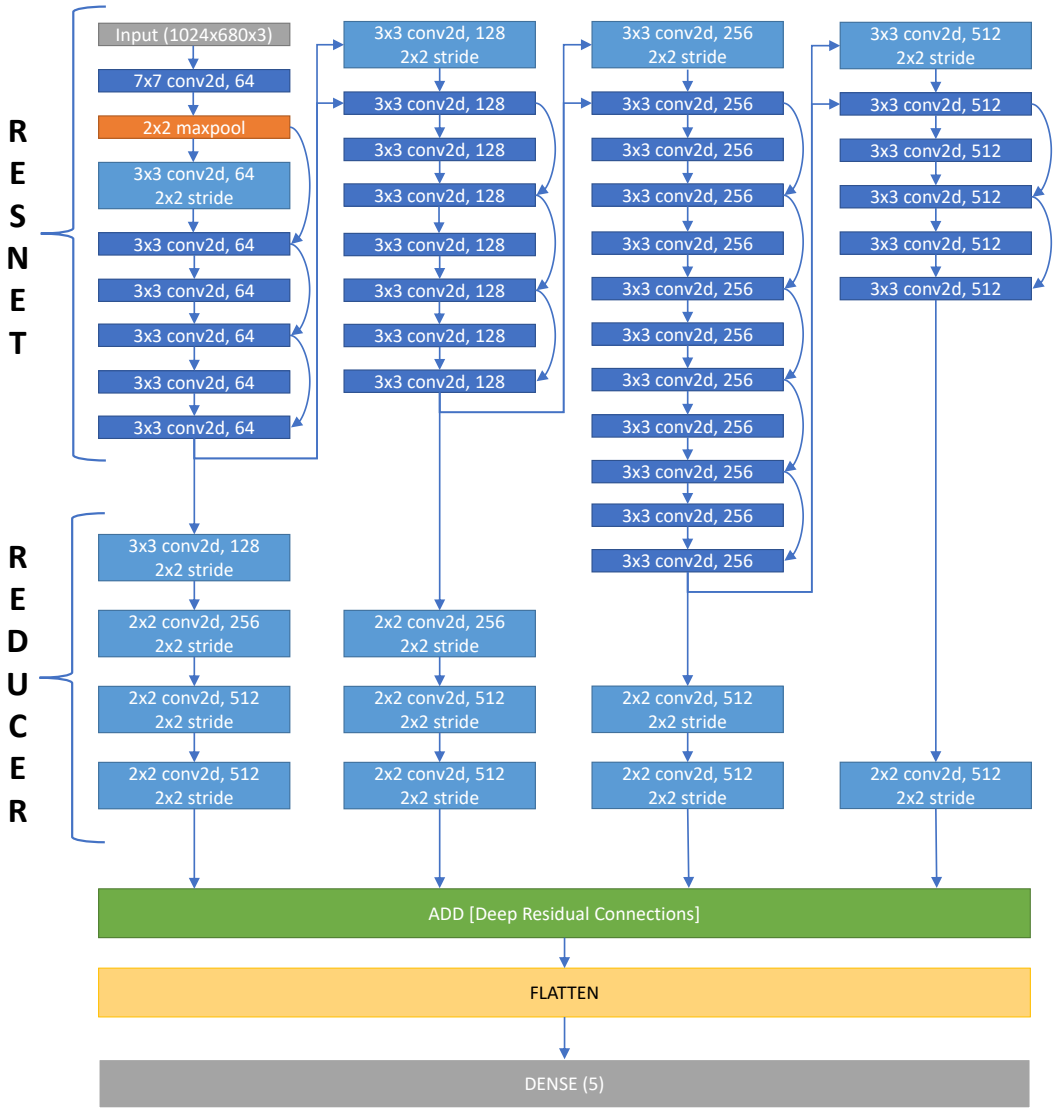


Figure 6.3: Convolutional Neural Network Architecture



Figure 6.4: (A) Human ranking = 0, Model Ranking = 1; (B) Human ranking = 3, Model Ranking = 2; (c) Human ranking = 2, Model Ranking = 3; (D) Human ranking = 3, Model Ranking = 3

CHAPTER 7
CLOSING REMARKS AND FUTURE WORKS

In this dissertation, we have shown that we can use deep learning, and in particular CNNs to detect a variety of image quality problems. We have further shown how these problems can negatively impact classification tasks by performing a case study on medical image diagnosis. The results showed that sensitivity is affected more negatively than specificity in the presence of quality issues such as Gaussian noise, salt and pepper, Poisson noise, and speckle noise. In some instances, sensitivity was affected by a factor 5, meaning that 5 times as many positive cases were incorrectly identified as negative.

The subject of image quality was further expanded upon by the introduction of a novel CNN which is capable of learning the aesthetic qualities of photographs and assign an aesthetic ranking to the image. The results show that using the proposed approach, ratings provided by the model were within one point of the human provided ranking in 91% of cases. This work has immense implications in a multitude of applications including social media, digital photography, and e-commerce. In social media, an aesthetic ranking model can be used to determine which image to upload. As a photographer is taking photos, he or she can be shown helpful hints regarding the aesthetic ranking of a particular frame. Using this information the photographer can choose to take a photo from a different angle or with different lighting. Lastly, in e-commerce, such a system could be used to identify a buyer's sense of style. This enables sellers to show items that are in a completely different category, and look completely unique, but they match the taste of the target user.

As a followup study, it would be interesting to investigate the relationship between

prediction accuracy and noise injected into the image. We observed that Gaussian noise, when added to images of skin lesions, the classification accuracy either was not affected, or went up slightly. This is counter intuitive and it would be interesting to examine the mechanism by which this performance improvement is judged.

BIBLIOGRAPHY

[1] Arash Golchubian, Oge Marques, and Mehrdad Nojoumian. "Photo quality classification using deep learning". In: *Multimedia Tools and Applications* 80.14 (2021), pp. 22193–22208.

[2] Arash Golchubian, Oge Marques, and Mehrdad Nojoumian. "Improving medical diagnosis classification accuracy through deep learning detection of poor quality images". In: (under preparation).

[3] Arash Golchubian, Maria Davis, Mehrdad Nojoumian, and Borko Furht. "Aesthetic ranking of photos using Deep Learning". In: *Multimedia Tools and Applications* (under submission, 2022).

[4] Xiaoou Tang, Wei Luo, and Xiaogang Wang. "Content-based photo quality assessment". In: *IEEE Transactions on Multimedia* 15.8 (2013), pp. 1930–1943.

[5] Hamid R Sheikh, Zhou Wang, Lawrence Cormack, and Alan C Bovik. *LIVE image quality assessment database release 2 (2005)*. 2005.

[6] Ibrahem Kandel, Mauro Castelli, and Aleš Popovič. "Comparative study of first order optimizers for image classification using convolutional neural networks on histopathology images". In: *Journal of imaging* 6.9 (2020), p. 92.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[8] Mohammad Sadegh Ebrahimi and Hossein Karkeh Abadi. "Study of residual networks for image recognition". In: *Intelligent Computing*. Springer, 2021, pp. 754–763.

[9] Fengxiang He, Tongliang Liu, and Dacheng Tao. "Why resnet works? residuals generalize". In: *IEEE transactions on neural networks and learning systems* 31.12 (2020), pp. 5349–5362.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.

[11] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[12] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[13] Michael J Burke, Lisa M Finkelstein, and Michelle S Dusig. "On average deviation indices for estimating interrater agreement". In: *Organizational Research Methods* 2.1 (1999), pp. 49–68.

[14] Jérôme Da Rugna and Hubert Konik. "Automatic blur detection for metadata extraction in content-based retrieval context". In: *Internet Imaging V*. Vol. 5304. International Society for Optics and Photonics. 2003, pp. 285–295.

[15] Renting Liu, Zhaorong Li, and Jiaya Jia. "Image partial blur detection and classification". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.

[16] Bolan Su, Shijian Lu, and Chew Lim Tan. "Blurred image region detection and classification". In: *Proceedings of the 19th ACM international conference on Multimedia*. ACM. 2011, pp. 1397–1400.

[17] Ke Gu, Guangtao Zhai, Weisi Lin, Xiaokang Yang, and Wenjun Zhang. "No-reference image sharpness assessment in autoregressive parameter space". In: *IEEE Transactions on Image Processing* 24.10 (2015), pp. 3218–3231.

[18] Samuel J Yang, Marc Berndl, D Michael Ando, Mariya Barch, Arunachalam Narayanaswamy, Eric Christiansen, Stephan Hoyer, Chris Roat, Jane Hung, Curtis T Rueden, et al. "Assessing microscope image focus quality with deep learning". In: *BMC bioinformatics* 19.1 (2018), pp. 1–9.

[19] Simone Bianco, Luigi Celona, Paolo Napoletano, and Raimondo Schettini. "On the use of deep learning for blind image quality assessment". In: *Signal, Image and Video Processing* 12.2 (2018), pp. 355–362.

[20] Pina Marziliano, Frederic Dufaux, Stefan Winkler, and Touradj Ebrahimi. "A no-reference perceptual blur metric". In: *Image processing. 2002. Proceedings. 2002 international conference on*. Vol. 3. IEEE. 2002, pp. III–III.

[21] Yun-Chung Chung, Jung-Ming Wang, Robert R Bailey, Sei-Wang Chen, and Shyang-Lih Chang. "A non-parametric blur measure based on edge analysis for image processing applications". In: *Cybernetics and Intelligent Systems, 2004 IEEE Conference on*. Vol. 1. IEEE. 2004, pp. 356–360.

[22] Hanghang Tong, Mingjing Li, Hongjiang Zhang, and Changshui Zhang. "Blur detection for digital images using wavelet transform". In: *Multimedia and Expo, 2004. ICME'04. 2004 IEEE International Conference on*. Vol. 1. IEEE. 2004, pp. 17–20.

[23] Elena Tsomko, Hyoung Joong Kim, Joonki Paik, and In-Kwon Yeo. "Efficient method of detecting blurry images". In: *Journal of Ubiquitous Convergence Technology* 2.1 (2008), pp–27.

[24] S Alireza Golestaneh and Lina J Karam. "Spatially-Varying Blur Detection Based on Multiscale Fused and Sorted Transform Coefficients of Gradient Magnitudes." In: *CVPR*. 2017, pp. 596–605.

[25] Yubin Deng, Chen Change Loy, and Xiaoou Tang. "Image aesthetic assessment: An experimental survey". In: *IEEE Signal Processing Magazine* 34.4 (2017), pp. 80–106.

[26] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z Wang. "Studying aesthetics in photographic images using a computational approach". In: *European conference on computer vision*. Springer. 2006, pp. 288–301.

[27] Shao-Fu Xue, Henry Tang, Dan Tretter, Qian Lin, and Jan Allebach. "Feature design for aesthetic inference on photos with faces". In: *2013 IEEE International Conference on Image Processing*. IEEE. 2013, pp. 2689–2693.

[28] Yan Ke, Xiaoou Tang, and Feng Jing. "The design of high-level features for photo quality assessment". In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 1. IEEE. 2006, pp. 419–426.

[29] Tunç Ozan Aydın, Aljoscha Smolic, and Markus Gross. "Automated aesthetic analysis of photographic images". In: *IEEE transactions on visualization and computer graphics* 21.1 (2014), pp. 31–42.

[30] Yiwen Luo and Xiaoou Tang. "Photo and video quality evaluation: Focusing on the subject". In: *European Conference on Computer Vision*. Springer. 2008, pp. 386–399.

[31] Luca Marchesotti, Florent Perronnin, Diane Larlus, and Gabriela Csurka. "Assessing the aesthetic quality of photographs using generic image descriptors". In: *2011 international conference on computer vision*. IEEE. 2011, pp. 1784–1791.

[32] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. "Improving the fisher kernel for large-scale image classification". In: *European conference on computer vision*. Springer. 2010, pp. 143–156.

[33] Hsiao-Hang Su, Tse-Wei Chen, Chieh-Chi Kao, Winston H Hsu, and Shao-Yi Chien. "Scenic photo quality assessment with bag of aesthetics-preserving features". In: *Proceedings of the 19th ACM international conference on Multimedia*. 2011, pp. 1213–1216.

[34] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012), pp. 1097–1105.

[35] Yi Sun, Xiaogang Wang, and Xiaoou Tang. "Hybrid deep learning for face verification". In: *Proceedings of the IEEE international conference on computer vision*. 2013, pp. 1489–1496.

[36] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. "Large-scale video classification with convolutional neural networks". In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2014, pp. 1725–1732.

[37] Wanli Ouyang, Xiaogang Wang, Xingyu Zeng, Shi Qiu, Ping Luo, Yonglong Tian, Hongsheng Li, Shuo Yang, Zhe Wang, Chen-Change Loy, et al. "Deepid-net: Deformable deep convolutional neural networks for object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 2403–2412.

[38] Zhe Dong and Xinmei Tian. "Multi-level photo quality assessment with multi-view features". In: *Neurocomputing* 168 (2015), pp. 308–319.

[39] Yueying Kao, Ran He, and Kaiqi Huang. "Visual aesthetic quality assessment with multi-task deep learning". In: *arXiv preprint arXiv:1604.04970* 5 (2016).

[40] Xin Lu, Zhe Lin, Hailin Jin, Jianchao Yang, and James Z Wang. "Rapid: Rating pictorial aesthetics using deep learning". In: *Proceedings of the 22nd ACM international conference on Multimedia*. 2014, pp. 457–466.

[41] Long Mai, Hailin Jin, and Feng Liu. "Composition-preserving deep photo aesthetics assessment". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 497–506.

[42] Gautam Malu, Raju S Bapi, and Bipin Indurkhya. "Learning photography aesthetics with deep cnns". In: *arXiv preprint arXiv:1707.03981* (2017).

[43] Xinmei Tian, Zhe Dong, Kuiyuan Yang, and Tao Mei. "Query-dependent aesthetic model with deep learning for photo quality assessment". In: *IEEE Transactions on Multimedia* 17.11 (2015), pp. 2035–2048.

[44] Shu Kong, Xiaohui Shen, Zhe Lin, Radomir Mech, and Charless Fowlkes. "Photo aesthetics ranking network with attributes and content adaptation". In: *European Conference on Computer Vision*. Springer. 2016, pp. 662–679.

[45] Zhangyang Wang, Shiyu Chang, Florin Dolcos, Diane Beck, Ding Liu, and Thomas S Huang. "Brain-inspired deep networks for image aesthetics assessment". In: *arXiv preprint arXiv:1601.04155* (2016).

[46] Xin Lu, Zhe Lin, Xiaohui Shen, Radomir Mech, and James Z Wang. "Deep multi-patch aggregation network for image style, aesthetics, and quality estimation". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 990–998.

[47] Zhe Dong, Xu Shen, Houqiang Li, and Xinmei Tian. "Photo quality assessment with DCNN that understands image well". In: *International Conference on Multimedia Modeling*. Springer. 2015, pp. 524–535.

[48] Abrar H Abdulnabi, Gang Wang, Jiwen Lu, and Kui Jia. "Multi-task CNN model for attribute prediction". In: *IEEE Transactions on Multimedia* 17.11 (2015), pp. 1949–1959.

[49] Rich Caruana. "Multitask learning". In: *Machine learning* 28.1 (1997), pp. 41–75.

[50] Leida Li, Hancheng Zhu, Sicheng Zhao, Guiguang Ding, and Weisi Lin. "Personality-assisted multi-task learning for generic and personalized image aesthetics assessment". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 3898–3910.

[51] Xin Lu, Zhe Lin, Hailin Jin, Jianchao Yang, and James Z Wang. "Rating image aesthetics using deep learning". In: *IEEE Transactions on Multimedia* 17.11 (2015), pp. 2021–2034.

[52] Yueying Kao, Chong Wang, and Kaiqi Huang. "Visual aesthetic quality assessment with a regression model". In: *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2015, pp. 1583–1587.

[53] Xin Jin, Jingying Chi, Siwei Peng, Yulu Tian, Chaochen Ye, and Xiaodong Li. "Deep image aesthetics classification using inception modules and fine-tuning connected layer". In: *2016 8th International Conference on Wireless Communications & Signal Processing (WCSP)*. IEEE. 2016, pp. 1–6.

[54] Xavier Moles Lopez, Etienne D'Andrea, Paul Barbot, Anne-Sophie Bridoux, Sandrine Rorive, Isabelle Salmon, Olivier Debeir, and Christine Decaestecker. "An automated blur detection method for histological whole slide imaging". In: *PloS one* 8.12 (2013), e82710.

[55] Ana Jiménez, Gloria Bueno, Gabriel Cristóbal, Oscar Déniz, David Toomey, and Catherine Conway. "Image quality metrics applied to digital pathology". In: *Optics, Photonics and Digital Technologies for Imaging Applications IV*. Vol. 9896. International Society for Optics and Photonics. 2016, 98960S.

[56] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. "Deep learning for computer vision: A brief review". In: *Computational intelligence and neuroscience* 2018 (2018).

[57] Sharib Ali, Felix Zhou, Adam Bailey, Barbara Braden, James E East, Xin Lu, and Jens Rittscher. "A deep learning framework for quality assessment and restoration in video endoscopy". In: *Medical Image Analysis* 68 (2021), p. 101900.

[58] Cheng Jiang, Jun Liao, Pei Dong, Zhaoxuan Ma, De Cai, Guoan Zheng, Yueping Liu, Hong Bu, and Jianhua Yao. "Blind deblurring for microscopic pathology images using deep learning networks". In: *arXiv preprint arXiv:2011.11879* (2020).

[59] Rafael Redondo, Gabriel Cristóbal, Gloria Bueno Garcia, Oscar Deniz, Jesus Salido, Maria del Milagro Fernandez, Juan Vidal, Juan Carlos Valdiviezo, Rodrigo Nava, Boris Escalante-Ramırez, et al. "Autofocus evaluation for brightfield microscopy pathology". In: *Journal of biomedical optics* 17.3 (2012), p. 036008.

[60] Attila Tiba, Zsombor Bartik, Henrietta Toman, and Andras Hajdu. "Detecting outlier and poor quality medical images with an ensemble-based deep learning system". In: *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*. IEEE. 2019, pp. 99–104.

[61] Mussarat Yasmin, Muhammad Sharif, and Sajjad Mohsin. "Neural networks in medical imaging applications: A survey". In: *World Applied Sciences Journal* 22.1 (2013), pp. 85–96.

[62] Caglar Senaras, M Khalid Khan Niazi, Gerard Lozanski, and Metin N Gurcan. "DeepFocus: detection of out-of-focus regions in whole slide digital images using deep learning". In: *PloS one* 13.10 (2018), e0205387.

[63] Gabriele Campanella, Arjun R Rajanna, Lorraine Corsale, Peter J Schüffler, Yukako Yagi, and Thomas J Fuchs. "Towards machine learned quality control: A benchmark for sharpness quantification in digital pathology". In: *Computerized Medical Imaging and Graphics* 65 (2018), pp. 142–151.

[64] Mark Brinded. "Computer Vision Methods for Detection of Blurry Photographs". PhD thesis. University of Leeds, School of Computing Studies, 2011.

[65] Ping Hsu and Bing-Yu Chen. "Blurred image detection and classification". In: *International Conference on Multimedia Modeling*. Springer. 2008, pp. 277–286.

[66] Wei Liu and Weisi Lin. "Additive white Gaussian noise level estimation in SVD domain for images". In: *IEEE Transactions on Image processing* 22.3 (2013), pp. 872–883.

[67] Wafaa Shihab Ahmed et al. "The Impact of Filter Size and Number of Filters on Classification Accuracy in CNN". In: *2020 International Conference on Computer Science and Software Engineering (CSASE)*. IEEE. 2020, pp. 88–93.

[68] Arash Golchubian, Oge Marquez, and Mehrdad Nojoumian. *Photo Quality Classification Using Deep Learning - Dataset and Programming*. 2020. URL: `https://github.com/agolchub/Photo%5C_Quality%5C_Classification`.

[69] Hamed Habibi Aghdam and Elnaz Jahani Heravi. "Guide to Convolutional Neural Networks". In: *New York, NY: Springer. doi* 10 (2017), pp. 225–226.

[70] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition". In: *Neural networks* 32 (2012), pp. 323–332.

[71] Hardeep Singh, Ashley ND Meyer, and Eric J Thomas. "The frequency of diagnostic errors in outpatient care: estimations from three large observational studies involving US adult populations". In: *BMJ quality & safety* 23.9 (2014), pp. 727–731.

[72] Muhammad Khalid Khan Niazi, Anil V Parwani, and Metin N Gurcan. "Digital pathology and artificial intelligence". In: *The lancet oncology* 20.5 (2019), e253–e261.

[73] Ziqi Tang, Kangway V Chuang, Charles DeCarli, Lee-Way Jin, Laurel Beckett, Michael J Keiser, and Brittany N Dugger. "Interpretable classification of Alzheimer's disease pathologies with a convolutional neural network pipeline". In: *Nature communications* 10.1 (2019), pp. 1–14.

[74] Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, et al. "Clinically applicable deep learning for diagnosis and referral in retinal disease". In: *Nature medicine* 24.9 (2018), pp. 1342–1350.

[75] Huiying Liang, Brian Y Tsui, Hao Ni, Carolina CS Valentim, Sally L Baxter, Guangjian Liu, Wenjia Cai, Daniel S Kermany, Xin Sun, Jiancong Chen, et al. "Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence". In: *Nature medicine* 25.3 (2019), pp. 433–438.

[76] Geert Litjens, Clara I Sánchez, Nadya Timofeeva, Meyke Hermsen, Iris Nagtegaal, Iringo Kovacs, Christina Hulsbergen-Van De Kaa, Peter Bult, Bram Van Ginneken, and Jeroen Van Der Laak. "Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis". In: *Scientific reports* 6.1 (2016), pp. 1–11.

[77] Kun-Hsing Yu, Andrew L Beam, and Isaac S Kohane. "Artificial intelligence in healthcare". In: *Nature biomedical engineering* 2.10 (2018), pp. 719–731.

[78] Barmak Honarvar Shakibaei Asli, Yifan Zhao, and John Ahmet Erkoyuncu. "Motion Blur Invariant for Estimating Motion Parameters of Medical Ultrasound Images". In: (2021).

[79] Navid Farahani, Anil V Parwani, and Liron Pantanowitz. "Whole slide imaging in pathology: advantages, limitations, and emerging perspectives". In: *Pathology and Laboratory Medicine International* 7 (2015), pp. 23–33.

[80] Kangkana Bora, Manish Chowdhury, Lipi B Mahanta, Malay Kumar Kundu, and Anup Kumar Das. "Automated classification of Pap smear images to detect cervical dysplasia". In: *Computer methods and programs in biomedicine* 138 (2017), pp. 31–47.

[81] Eralda Nishani and Betim Çiço. "Computer vision approaches based on deep learning and neural networks: Deep neural networks for video analysis of human pose estimation". In: *2017 6th Mediterranean Conference on Embedded Computing (MECO)*. IEEE. 2017, pp. 1–4.

[82] Waseem Rawat and Zenghui Wang. "Deep convolutional neural networks for image classification: A comprehensive review". In: *Neural computation* 29.9 (2017), pp. 2352–2449.

[83] Nitish Shirish Keskar and Richard Socher. "Improving generalization performance by switching from adam to sgd". In: *arXiv preprint arXiv:1712.07628* (2017).

[84] Pan Zhou, Jiashi Feng, Chao Ma, Caiming Xiong, Steven Chu Hong Hoi, et al. "Towards theoretically understanding why sgd generalizes better than adam in deep learning". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 21285–21296.

[85] Yueying Kao, Ran He, and Kaiqi Huang. "Deep aesthetic quality assessment with semantic information". In: *IEEE Transactions on Image Processing* 26.3 (2017), pp. 1482–1495.

[86] Naila Murray, Luca Marchesotti, and Florent Perronnin. "AVA: A large-scale database for aesthetic visual analysis". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2012, pp. 2408–2415.

[87] Jose Llamas, Pedro M Lerones, Roberto Medina, Eduardo Zalama, and Jaime Gómez-Garcıa-Bermejo. "Classification of architectural heritage images using deep learning techniques". In: *Applied Sciences* 7.10 (2017), p. 992.

[88] Maher Ibrahim Sameen, Biswajeet Pradhan, and Omar Saud Aziz. "Classification of very high resolution aerial photos using spectral-spatial convolutional neural networks". In: *Journal of Sensors* 2018 (2018).

[89] Hanfa Xing, Yuan Meng, Zixuan Wang, Kaixuan Fan, and Dongyang Hou. "Exploring geo-tagged photos for land cover validation with deep learning". In: *ISPRS journal of photogrammetry and remote sensing* 141 (2018), pp. 237–251.

[90] Guang Xu, Xuan Zhu, Dongjie Fu, Jinwei Dong, and Xiangming Xiao. "Automatic land cover classification of geo-tagged field photos by deep learning". In: *Environmental Modelling & Software* 91 (2017), pp. 127–134.

[91] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. "Striving for simplicity: The all convolutional net". In: *arXiv preprint arXiv:1412.6806* (2014).

[92] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. "Language models are few-shot learners". In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[93] Hao-Wen Dong and Yi-Hsuan Yang. “Convolutional generative adversarial networks with binary neurons for polyphonic music generation”. In: *arXiv preprint arXiv:1804.09399* (2018).

[94] Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. “Can: Creative adversarial networks, generating” art” by learning about styles and deviating from style norms”. In: *arXiv preprint arXiv:1706.07068* (2017).