University of Windsor

Scholarship at UWindsor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2-15-2024

Enhancing Marijuana Intoxication Detection by using Deep Learning-based Architecture and Image Augmentation

Puneet Jain
University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Part of the Computer Sciences Commons

Recommended Citation

Jain, Puneet, "Enhancing Marijuana Intoxication Detection by using Deep Learning-based Architecture and Image Augmentation" (2024). *Electronic Theses and Dissertations*. 9445.
https://scholar.uwindsor.ca/etd/9445

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Enhancing Marijuana Intoxication Detection by using Deep Learning-based Architecture and Image Augmentation

By

Puneet Jain

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2024

©2024 Puneet Jain

Enhancing Marijuana Intoxication Detection by using Deep Learning-based Architecture and Image Augmentation

by

Puneet Jain

APPROVED BY:

H. Wu
Department of Economics

I. Ahmad
School of Computer Science

D. Wu
School of Computer Science

January 22, 2024

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

The primary psychoactive component of marijuana is Δ-9-tetrahydrocannabinol (THC). There has been a significant increase in motor vehicle accidents and workplace mishaps due to the misuse of marijuana, often leading to intoxication impacting societies worldwide. Civil bodies and organizations continue to rely on conventional marijuana intoxication detection techniques to battle such problems. They often employ techniques such as field sobriety tests, breath analyzer tests, blood tests and DRUID. These tests for detecting cannabis use have demonstrated a range of limitations. Consequently, the emphasis is directed toward developing a machine learning-based solution that can reliably and instantaneously determine whether a person is under the influence of marijuana.

Developing a machine-learning solution for marijuana detection requires extensive, credible data for training, and the scarcity of such data shows the need for improved data generation and classification methods. Recent work addresses the issue of data availability by sourcing images of marijuana-intoxicated individuals from YouTube and Google searches. Sourced images were used to train MobileNet, SVM, Decision Tree and Random Forest classifier, which detects the presence of marijuana. However, the recent work must incorporate current state-of-the-art neural classification models and deep learning-based image augmentation techniques. This research implements StlyeGAN3, a state-of-the-art model for image generation, to proliferate the dataset of screenshots of faces of marijuana-intoxicated individuals sourced from the internet. Additionally, ResNet-50, InceptionV3 and VGG-16 classifiers were used to detect marijuana intoxication. VGG-16 classifiers outperformed other classifiers and achieved an accuracy of 94.66%, precision of 96.84%, recall of 89.32%, and an F1-score of 92.92%, surpassing recent work.

DEDICATION

I dedicate this thesis to my mom and dad for their incredible love and support. Their utmost love and support throughout my career and their selfless commitment to giving me the future I always wanted have provided me with the necessary boost I needed in my life to achieve success. Furthermore, I dedicate it to my uncle, Anoop Jain and brother, Lovish Jain, for being ever so lovable, kind, supportive, and the backbone of what I am today. I want to thank Karan Saxena, Jess Joesph Benny and Tanmany Daulatwal for supporting and pushing me during my thesis.

ACKNOWLEDGEMENTS

I would like to sincerely express my most profound gratitude towards my supervisor, Dr. Dan Wu, whose input helped me immensely. With his input, I could look at my research with a different perspective and a more critical eye. Secondly, I would like to express my gratitude to my thesis committee members for their beneficial advice and suggestions for my thesis. I humbly thank the School of Computer Science and everyone who helped me in this regard.

TABLE OF CONTENTS

DECLARATION OF ORIGINALITY **III**

ABSTRACT **IV**

DEDICATION **V**

LIST OF TABLES **IX**

LIST OF FIGURES **X**

LIST OF ABBREVIATIONS **XI**

1 Introduction **1**
- 1.1 What is Marijuana and its uses? 1
- 1.2 Impact on the Human Body 2
- 1.3 Impact of Cannabis on the economy 3
- 1.4 Detection of Marijuana Intoxication 5
  - 1.4.1 Field Sobriety Tests and Drug Recognition Experts (DREs) . 5
  - 1.4.2 DRUID Project 6
  - 1.4.3 Oral Fluid Testing 6
  - 1.4.4 Breath Testing 7
- 1.5 Marijuana Intoxication Detection Using Machine Learning 7
- 1.6 Thesis Contributions 9
- 1.7 Thesis Outline 11

2 Relevant Background **12**
- 2.1 Convolutional Neural Network 12
- 2.2 Image Augmentation 18
  - 2.2.1 GAN 18
  - 2.2.2 StyleGAN 20
  - 2.2.3 StyleGAN3 23
- 2.3 CNN Architectures 27
  - 2.3.1 VGG-16 27
  - 2.3.2 Inception-v3 29
  - 2.3.3 ResNet-50 33

3 Literature Review **35**
- 3.1 Field Sobriety Tests and Drug Recognition Experts (DREs) 35
- 3.2 DRUID Project 36
- 3.3 Oral Fluid Testing 37
- 3.4 Breath Testing 37
- 3.5 Marijuana intoxication detection using Machine Learning 37

4 Motivation and Research Objective 41
4.1 Motivation . . . 41
4.2 Research Objective . . . 42

5 Expansion of Bloodshot Eyes Dataset 44
5.1 Dataset and Evaluation Metric . . . 44
5.2 Flow for Dataset Creation . . . 46
  5.2.1 Dataset Pre-processing . . . 47
  5.2.2 Augmentation of Images via StyleGAN3 . . . 48
5.3 Results and Discussion . . . 50

6 Enhancement of Marijuana Intoxication Detection Models 53
6.1 Dataset and Evaluation Metrics . . . 53
6.2 Pipeline for Fine-tuning CNN Models . . . 56
  6.2.1 Model Setup . . . 57
  6.2.2 Tuning Model . . . 60
6.3 Results and Discussion . . . 61

7 Conclusion and Future Work 65
7.1 Conclusion . . . 65
7.2 Future Works . . . 66

REFERENCES 67

VITA AUCTORIS 73

LIST OF TABLES

1 FID for various configurations of StyleGAN2 and StyleGAN3. 26

2 MID2021 Summary 45

3 MID2023 Summary 54

4 Best Hyper-parameters for models trained on MID2023 62

5 Best Performance for Marijuana Intoxication Models 64

LIST OF FIGURES

1 Dlib Facial Landmarks [30]. 8

2 An Artificial Neural Network [46]. 13
3 Family of 3 continuous piecewise linear functions for ten parameters θ [47]. 14
4 1D convolution with kernel size three and stride one. a, b, and c have zero padding, while d has no padding [48]. 16
5 2D convolution with kernel size of 3×3 applied to an RGB image [49]. 17
6 GAN training on 1D examples [50]. 19
7 ProGAN Generator vs StyleGAN Generator [23]. 21
8 VGG-16 Architecture [19]. 28
9 Factorization Example [56]. 30
10 Inception V3 Architecture [56]. 31
11 ResNet-50 Architecture [16]. 33

12 Pre-processed image from MID2021 and CelebA. 47
13 Tick vs FID in StyleGAN3. 51
14 Marijuana intoxicated eyes generated by StyleGAN3. 51
15 Images generated by StyleGAN3-R at 2400 kimg. 52

16 Pipeline for fine-tuning marijuana intoxication detection models. . . 56
17 Accuracy and Loss over Epochs for best MobileNet Model on MID2023. 61
18 Accuracy and Loss over Epochs for best ResNet-50 on MID2023. . . 62
19 Accuracy and Loss over Epochs for best VGG-16 on MID2023. . . . 63
20 Accuracy and Loss over Epochs for best Inception-v3 on MID2023. . 63

LIST OF ABBREVIATIONS

PTSD	Post-traumatic stress disorder
TP	True Positive
FN	False Negative
FP	False Positive
TN	True Negative
THC	Δ-9-tetrahydrocannabinol
CBD	Cannabidiol
HIV	Human Immunodeficiency Viruses
AIDS	Acquired ImmunoDeficiency Syndrome
FDA	Food and Drug Administration
NIDA	National Institute on Drug Abuse
ABCD	Adolescent Brain Cognitive Development
DRE	Drug Recognition Expert
FST	Field Sobriety Test
SFTS	Standardized Field Sobriety Test
HGN	Horizontal Gaze Nystagmus
WAT	Walk and Turn
OLS	One Leg Stand
FTN	Finger to Nose (FTN)
MRB	Modified Romberg Balance
DUI	Driving under the influence

DRUID	Driving under the Influence of Drugs, Alcohol and Medicines
CNN	Convolutional Neural Network
SVM	Support Vector Machines
GAN	General Adversarial Network
ProGAN	Progressive General Adversarial Network
NN	Neural Network
ReLU	Rectified Linear Unit
MID2021	Marijuana Intoxicated Dataset 2021
MID2023	Marijuana Intoxicated Dataset 2023

CHAPTER 1

Introduction

1.1 What is Marijuana and its uses?

Cannabis, also referred to as marijuana, is a psychoactive substance derived from the cannabis plant [2]. The principal psychoactive compound found in cannabis is Δ-9-tetrahydrocannabinol (THC), which is just one of the 483 known constituents within the plant. Among these constituents are at least 65 other cannabinoids, including cannabidiol (CBD). Cannabis is employed for both recreational and medicinal purposes. Its legality varies among over 40 countries, with potential divergent effects on physical and mental well-being.

- **Recreational Usage**: The majority of cannabis products are derived from the flowers and leaves of the cannabis plant, and their THC potency can vary significantly based on the production process. The various forms of marijuana are fresh or dried leaves, cannabis oil, chemically concentrated extracts, edibles, tinctures and creams. The range of THC potency across these various forms is quite extensive, spanning from up to 3% in cannabis oil to as high as 90% in chemically concentrated extracts. This substantial variability carries implications for consumers, medical practitioners, and regulatory bodies alike, as it underscores the potential for a wide spectrum of psychoactive effects associated with different products.

- **Medical Applications**: Cannabis-based medicine has demonstrated potential utility in specific medical conditions. As indicated in a study by Azcarate et al. [8], prevalent medical motives for marijuana use encompass anxiety (49%), insomnia (47%), chronic pain (42%), and depression (39%). Additionally, individuals reported using cannabis for purposes such as mood stabilization, arthritis, migraines, post-traumatic stress disorder (PTSD), alleviating cancer symptoms, enhancing libido, managing glaucoma, mitigating seizures/epilepsy, addressing multiple sclerosis, and dealing with HIV/AIDS. Notably, the study found gender-specific variations, with women more likely than men to employ marijuana for managing PTSD, insomnia, anxiety, and migraines. In contrast, men demonstrated a greater propensity for its use in mood stabilization.

1.2 Impact on the Human Body

The effects of marijuana on the human body can be both positive [2], such as pain relief and relaxation, and harmful, including lung irritation and impaired memory. These effects can vary based on factors such as the amount, method, and frequency of use, as well as an individual's sensitivity and tolerance. In this section, we will explore the impact of marijuana on both the brain and other parts of the body, which might lead to potential costs and harms associated with its misuse [14].

- **Marijuana's Impact on the Brain**: Marijuana's active component, THC, interacts with brain cell receptors that are responsive to natural THC-like chemicals, integral to normal brain development and function. The overstimulation of these receptors by marijuana leads to the signature high, along with various short-term and long-term effects. The short-term effects of its consumption induce alterations in sensory and time perception, mood swings, cognitive and motor skill impairments, and, at high doses, hallucinations and increased psychosis risk [2]. These effects are intensified with the regular use of high-potency marijuana. Long-term consumption could start during adolescence, leading to impaired thinking, memory, and learning due to alterations in brain connectivity. A study revealed a notable decline in IQ points in individuals with persistent marijuana use from their teens, with incomplete recovery even after cessation [39] [2]. Additionally, research implies that observed declines in IQ may be attributed to shared familial influences such as genetics and family environment [24]. Ongoing research and longitudinal studies like the Adolescent Brain Cognitive Development (ABCD) study aim to offer deeper insights into marijuana's long-term effects on the developing brain.

- **Marijuana's Effect on the Body**: The smoke from marijuana irritates the lungs, leading to symptoms similar to those seen in tobacco smokers. Frequent marijuana smokers may experience daily cough, increased production of phlegm, a higher risk of lung infections, and more frequent lung illnesses [2]. Marijuana use can elevate heart rate for up to three hours after smoking. This effect may increase the risk of a heart attack, particularly in older individuals and those with preexisting heart conditions [2]. Using marijuana during pregnancy is associated with lower birth weight and an increased risk of both brain and behavioural problems in babies. Marijuana use during pregnancy can affect specific developing parts of the fetus's brain. Children exposed to marijuana in the womb face an elevated risk of attention, memory, and problem-solving difficulties compared to unexposed children. Recent research also suggests that moderate amounts of THC can be excreted into the breast milk of nursing mothers [2], potentially affecting the developing brain of the baby. Additionally, there is an increased risk of preterm births. Regular, long-term marijuana use can lead to a condition known as Cannabinoid Hyperemesis Syndrome [2], characterized by recurring cycles of severe nausea, vomiting, and dehydration, sometimes requiring emergency medical attention.

1.3 Impact of Cannabis on the economy

A report published in 2023 by the Canadian Centre on Substance Use and Addiction (CCSA) undertook an extensive analysis of the costs and detrimental effects associated with substance use in Canada from 2007 to 2020 [14]. This comprehensive report provides insights into the economic and societal implications of cannabis consumption and compares it to other substances, including alcohol, tobacco, cocaine, and opioids. The analysis encompasses various facets such as productivity, healthcare, criminal justice, and other direct expenses. These include costs related to federal funding for research and prevention initiatives, fire-related damage, motor vehicle accidents attributed to substance use, workplace drug testing, social assistance, employee assistance programs, and workers' compensation.

The findings from this report [14] indicate that in 2020, cannabis contributed to 4.9% of the total costs associated with substance use in Canada, amounting to CAD 2.4$ billion. In contrast, alcohol constituted 40.1% of the total expenses, equaling CAD 19.7$ billion, while tobacco represented 22.7% of the costs, totalling CAD 11.2$ billion. Cocaine accounted for 8.5% of the charges, equivalent to CAD 4.2$ billion, while opioids constituted 14.4% of the expenses, totalling 7.1 billion CAD.

The majority of the expenses associated with cannabis use primarily resulted from the criminal justice system (45%), followed by healthcare (20%), other direct costs (19%), and lost productivity (16%). In contrast, alcohol exhibited a similar distribution of costs, with lost productivity accounting for 40.8% of the total expenses, healthcare for 31.8%, criminal justice for 20%, and other direct costs for 8.2%. However, tobacco showed a distinct cost distribution, with healthcare expenses comprising 48.5%

Although the overall costs associated with alcohol, tobacco, opioids, and cocaine are significantly higher compared to cannabis, it's noteworthy that the expenses related to motor vehicle accidents and workplace drug testing for cannabis are comparable to those of alcohol. These findings shed light on the multifaceted economic impact of cannabis use in Canada, which makes marijuana detection an essential area of study. The following sub-section will discuss marijuana intoxication detection strategies.

1.4 Detection of Marijuana Intoxication

Marijuana intoxication detection involves identifying whether an individual is under the influence of marijuana. Various methods, such as urine, blood, and saliva tests, are employed to detect the presence of THC, the psychoactive component of cannabis. In Canada, expressly prohibited levels of THC, ranging from 2 ng to 5 ng per ml of blood, have been established, with possession above these levels considered a severe offence [41].

Impaired driving due to elevated levels of THC or other impairing substances is a severe criminal offence with substantial public safety implications. The penalties for such violations vary depending on factors like the concentration of drugs, whether it is a repeat offence, and if it has caused bodily harm or death.

To assess marijuana intoxication, various methods evaluating reaction time, decision-making abilities, hand-eye coordination, and observation of physical symptoms like red eyes are used [38]. These methods, including the application of machine learning techniques, could be essential tools for law enforcement and workplace testing to detect marijuana intoxication effectively.

1.4.1 Field Sobriety Tests and Drug Recognition Experts (DREs)

The initial assessment methods for suspected drug-induced impairment, especially in roadside situations, often involve field sobriety tests (FSTs). These tests typically comprise a series of physical and cognitive tasks designed to evaluate potential impairment. The Standardized Field Sobriety Test (SFST) is a battery of tests employed by law enforcement officers to determine if an individual suspected of impaired driving is influenced by alcohol or other drugs. The SFST consists of three tests: the Horizontal Gaze Nystagmus (HGN) test, the Walk and Turn (WAT) test, and the One Leg Stand (OLS) test. In addition to the SFST, other tests can be utilized to identify drug impairment, such as the Finger to Nose (FTN) test and the Modified Romberg Balance (MRB) test.

To enhance the reliability of FSTs, law enforcement agencies utilize Drug Recognition Experts (DREs). DREs are officers who have undergone extensive training to identify drug-induced impairment. Their evaluations follow a standardized 12-step process, including examining the suspect's medical history, vital signs, eye movements, and performance on psychophysical tests.

1.4.2 DRUID Project

The DRUID (Driving under the Influence of Drugs, Alcohol and Medicines) project is a large-scale European initiative aimed at combating drug-impaired driving. It provides significant resources and guidelines for Drug Recognition Expert (DRE) evaluations. The project involves multiple research institutions, universities, and public interest groups collaborating to collect and analyze substance use and driving data.

The DRUID app was developed to detect and measure an individual's cognitive and psychomotor capabilities. Each DRUID test collects several hundred measurements of key neurophysiological indicators, including reaction time, hand-eye coordination, decision-making, time estimation, and balance. These measures are integrated into an algorithm that scores individuals based on established cognitive and motor impairment indicators. Higher DRUID scores indicate impairment, while lower scores indicate less or no impairment.

One of the key outcomes of the DRUID project is the standardization of DRE evaluations. The project provides extensive resources and guidelines for DREs, which have improved the consistency and reliability of DRE evaluations across Europe.

1.4.3 Oral Fluid Testing

Oral fluid testing is a non-invasive method used to detect cannabis use by analyzing an individual's oral fluid (saliva) for the presence of THC and other cannabinoids. A systematic review of the correlation between oral fluid and blood THC concentration found that while a correlation exists, it is not consistent or strong enough to precisely estimate blood THC concentration based solely on oral fluid tests.

Furthermore, oral fluid testing can sometimes yield false positives due to the presence of THC in the mouth immediately after smoking or consuming cannabis, even when the person is not impaired. This suggests that while oral fluid testing can indicate cannabis use, it may not be sufficient on its own to determine impairment.

1.4.4 Breath Testing

Breath testing is a relatively new method for detecting cannabis use by analyzing an individual's breath for specific volatile organic compounds (VOCs) associated with cannabis use. A pilot study has developed a comprehensive breath test capable of confirming recent cannabis use within the impairment window. However, this method is still in the early stages of development, and further studies are needed to validate its effectiveness.

Additionally, the breath test was designed for detecting inhaled cannabis, and its effectiveness in detecting cannabis use through other routes of administration remains unclear.

1.5 Marijuana Intoxication Detection Using Machine Learning

In identifying marijuana intoxication, the methods discussed thus far have relied on various approaches such as Field Sobriety Tests (FST), Drug Recognition Experts (DRE), assessments of cognitive and psychomotor abilities, oral fluid testing, or breath analysis. However, a novel approach was developed in [20] by harnessing the power of machine learning to detect the presence of marijuana. This methodology involved creating a specialized dataset comprising individuals who exhibited signs of marijuana intoxication, focusing on discerning red or bloodshot eyes as an indicator of marijuana consumption.

The dataset was curated by sourcing images of marijuana-intoxicated faces from

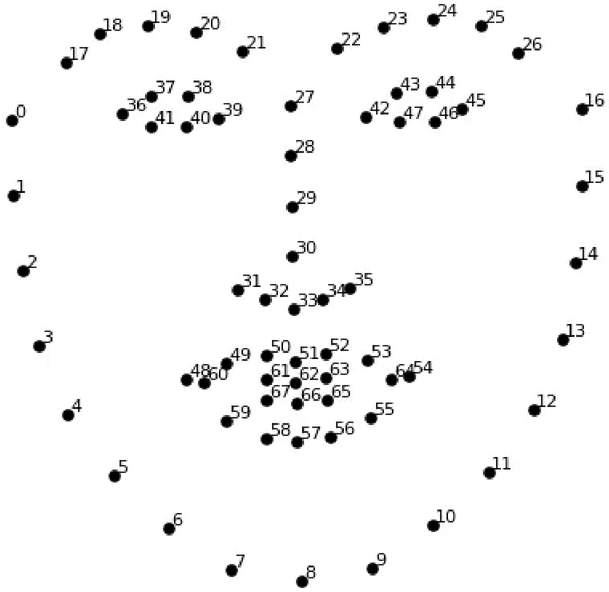


Fig. 1: Dlib Facial Landmarks [30].

online platforms, including Google Search and YouTube, and by applying traditional augmentation to collected data. This dataset will be referred to as the Marijuana Intoxicated Dataset 2021 (MID2021) in this thesis. MID2021 is a collection of 2750 images, including 600 original eye images sourced from the internet and 2150 images created by using traditional augmentation on original eye images. This dataset is equally split into two categories: 1375 images of bloodshot eyes, suggestive of marijuana intoxication, and 1375 images of sober eyes. It is important to note that while MID2021 focused on the eyes of marijuana-intoxicated individuals and Gadhiya [20] leveraged the facial landmark detector provided by the dlib library to isolate the relevant feature - eyes.

Dlib's facial landmark detector combines classical computer vision techniques and modern deep learning methods, including convolutional neural networks (CNNs). Dlib offers several facial landmark detectors, including the 68 landmark points detector, as shown in Figure 1. This figure shows that 68 landmark points correspond to distinct facial features, including eye corners, mouth corners, the tip of the nose, and facial contours. Moreover, it provides a more concise set of 5 landmark points, mainly eye corners and the nose tip. Additionally, dlib boasts a pre-trained face recognition model powered by CNN. This model excels in recognizing faces within images or video, exhibiting remarkable resilience to changes in illumination, pose, and facial expressions. Specific detection points are employed to detect eyes: points 36 to 41 pertain to the left eye, points 42 to 47 correspond to the right eye and points 48 to 67 correspond to the lips.

Traditional image augmentation techniques were applied to enhance the dataset's robustness and diversity. These techniques encompassed operations like image rotation, random cropping, flipping, and adjustments to image contrast. This augmentation process expanded the MID2021 to include 1375 images of intoxicated individuals and 1375 images depicting sober individuals.

Once the dataset was assembled and augmented, Gadhiya [20] employed a range of machine-learning algorithms to discern marijuana intoxication in individuals. These algorithms encompassed Support Vector Machines (SVM), Random Forests, Decision Trees, and MobileNet [20]. The performance metrics for various models in [20] revealed that the MobileNet outperformed others with a precision of 80%, recall of 83%, an F1-score of 82%, and an accuracy of 82%. The Support Vector Machine (SVM) demonstrated a precision of 63%, but its recall lagged at 56%, leading to an F1-score of 56% and an accuracy of 65%. The Decision Tree and Random Forest show similar accuracies of 65% and 72%, respectively. However, the Random Forest edges out with a slightly higher precision of 73% compared to the Decision Tree's 66%. Overall, the MobileNet emerges as the most efficient model.

1.6 Thesis Contributions

As elucidated in the preceding subsection, the foundational work in [20] utilized traditional augmentation to increase the MID2021 and MobileNet to detect marijuana intoxication. The main contributions of this research are summarized as follows.

- Deep learning-based augmentation techniques have gained significant attention in various fields, including medical imaging, skin disease diagnosis, character animation, style transfer, and image recognition. These techniques aim to enhance the performance and robustness of deep learning models by generating synthetic data or modifying existing data to increase the diversity and variability of the training set. The first contribution of this thesis is to improve image augmentation in [20] by integrating deep learning-based image augmentation methods, such as StyleGAN3 (elaborated in Section 2.4.4), to substantially augment the dataset. The better disentanglement of latent spaces facilitates the fine-grained editing of images in StyleGAN3. Better disentanglement means that different aspects of the eyes (like vein, pupil, iris) are more independently controlled. Moreover, StyleGAN3's capability for fine-grained editing supports making changes with a high level of detail and clarity, ensuring that even minor details can be tweaked precisely. When generating images of eyes, such control can enable specific adjustments to details like iris patterns or reflections without affecting other aspects of the eye. StyleGAN3 variants can generate images of eyes which will maintain their realism even when the eyes' position changes or when they're seen from different angles. This is crucial for generating varied images of eyes that look natural in different facial expressions and orientations. This augmentation proliferates the MID2021 with many diverse instances to generate a new dataset. This dataset will be called the Marijuana Intoxicated Dataset 2023 (MID2023). MID2023 consists of 4800 images, including 600 original eye images sourced from the original eye images in MID2021, 1200 images generated by StyleGAN3 and 3000 images generated by using traditional augmentation on both the original eye images and images generated by StyleGAN3. This dataset is equally split into two categories: 2400 images of bloodshot eyes, suggestive of marijuana intoxication, and 2400 images of sober eyes.

- MobileNet was used in [20] as an efficient marijuana intoxication model, outperforming classifiers such as SVM, Decision Tree and Random Forest when trained on MID2021. MobileNet is based on a streamlined architecture that can build lightweight deep neural networks. However, MobileNet was neither adequately deep nor sufficiently broad to proficiently extract discernible features from images of marijuana intoxication, especially those of the eyes. The second contribution of this thesis is to use more profound networks like VGG-16, ResNet-50, and Inception-v3 as these networks perform well for eyes disease-related datasets [11] [37]. The efficacy of the marijuana-intoxication detection model saw enhancements across CNN networks such as VGG-16, Inception-v3, and ResNet-50 when trained on the MID2023. For the MID2023 dataset, the VGG-16 model showcased superior performance with an accuracy of 94.66%, precision of 96.84%, recall of 89.32%, and an F1-score of 92.92%. ResNet-50 displayed lower metrics with an accuracy of 82.34%, precision of 71.49%, recall of 79.12%, and an F1-score of 75.11%. Inception-v3 achieved an accuracy of 90.18%, precision of 82.1%, recall of 95.12%, and an F1-score of 88.13%. VGG-16 stood out with the highest accuracy and F1-score for the MID2023 dataset.

1.7 Thesis Outline

The relevant background required for the thesis is discussed in Chapter 2, while Chapter 3 provides a literature review of relevant publications for marijuana intoxication detection. Chapter 4 describes the two research objectives for enhancing the detection of marijuana intoxication in [20] based on the two thesis contributions in section 1.6. In Chapter 5, the detailed technique involving the development of the new dataset MID2023 is thoroughly discussed. Chapter 6 will provide an in-depth discussion of the experimental methodologies for fine-tuning VGG-16, ResNet-50, Inception-v3 and MobileNet. Moreover, in Chapter 6, their performance will be evaluated based on performance metrics such as precision, recall, accuracy and F1-score. Chapter 7 will thoroughly present the conclusions and evaluations.

CHAPTER 2

Relevant Background

Chapter 1 explained how marijuana can affect our body and economy and discussed different ways to detect marijuana intoxication. Moreover, how marijuana intoxication detection [20] has been done by using machine learning was discussed in the previous chapter. This section contains background knowledge and terminology related to this thesis. In section 2.1, we will discuss the Convolutional Neural Network architecture components. Section 2.2 will discuss how to proliferate the dataset using deep learning-based image augmentation. In section 2.3, we will discuss different CNN architectures for detecting red eyes.

2.1 Convolutional Neural Network

Before moving on to convolutional neural networks (CNN), let us discuss the Artificial Neural Network (ANN). An ANN is composed of nodes or neurons interconnected in a structure similar to the human brain. These neurons are organized in layers: an input layer to receive the data, one or more hidden layers to process the data, and an output layer to deliver the final result. Each neuron in these layers acts as a basic computational unit. Each neuron computes a weighted sum of its inputs, where the weights represent the strength or influence of the connections. The neuron passes the weighted sum of its inputs through an activation function. This function determines whether and how much the signal should progress through the network. Common activation functions include sigmoid, hyperbolic tangent (tanh), and Rectified Linear Unit (ReLU). The choice of activation function is essential as it influences the net-

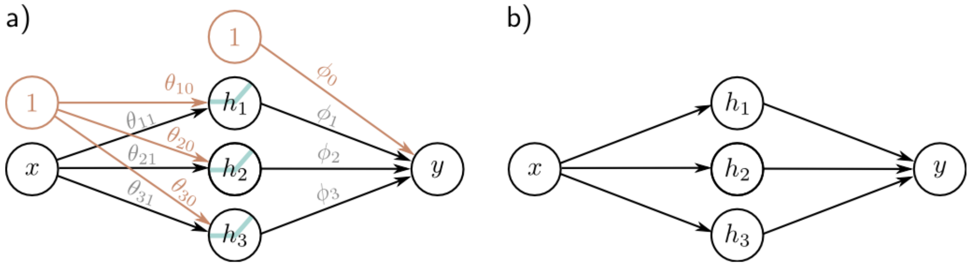


Fig. 2: An Artificial Neural Network [46].

work's ability to handle non-linear relationships and contributes to the ANN's overall learning and generalization capabilities. The general definition of an artificial neural network can be translated as follows:

$$\mathbf{y} = f[\mathbf{x}, \theta] \tag{1}$$

where: $\mathbf{y}$ is the multi-dimensional output in $\mathbb{R}^{D_o}$, and D_o is the number of dimensions of the $\mathbf{y}$. $\mathbf{x}$ is the multi-dimensional input in $\mathbb{R}^{D_i}$, and D_i is the number of dimensions of the $\mathbf{x}$. h is the hidden units in $\mathbb{R}^{D}$, and D represents the hidden units in the hidden layer. Each hidden unit h_d is computed as:

$$h_d = a\left[\theta_{d0} + \sum_{i=1}^{D_i} \theta_{di} x_i\right] \tag{2}$$

Here, a represents the activation function, θ_{d0} is the bias term for the d-th hidden unit, and θ_{di} are the weights associated with the i-th input for the d-th hidden unit.

Let's discuss an example of a neural network with one input, one output, and three hidden units as shown in Figure 2. The neural network $f[\mathbf{x}, \theta]$ that maps a scalar input $\mathbf{x}$ to a scalar output $\mathbf{y}$ with ten parameters $\theta = \{\theta_0, \theta_1, \theta_2, \theta_3, \theta_{10}, \theta_{11}, \theta_{20}, \theta_{21}, \theta_{30}, \theta_{31}\}$. Each of the ten parameters in this network is symbolized by a connection. Figure 2a shows how ten parameters are arranged in a neural network. For clarity, intercept parameters are usually omitted in visual representations, resulting in a more streamlined depiction like the one seen in Figure 2b. Based on the ten parameters, the ANN in Figure 2 could be written as below:

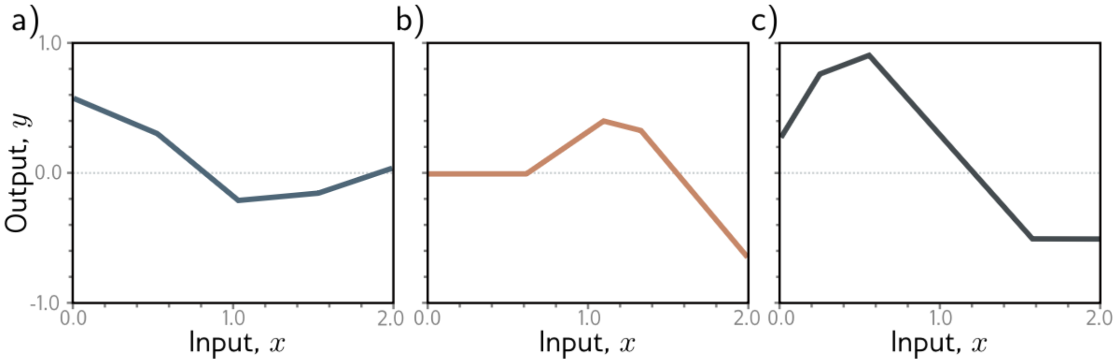


Fig. 3: Family of 3 continuous piecewise linear functions for ten parameters θ [47].

$$\mathbf{y} = \theta_0 + \theta_1 a[\theta_{10} + \theta_{11}\mathbf{x}] + \theta_2 a[\theta_{20} + \theta_{21}\mathbf{x}] + \theta_3 a[\theta_{30} + \theta_{31}\mathbf{x}] \tag{3}$$

The choice of an activation function a[•] could vary depending upon the problem statement. A common choice is the rectified linear unit (ReLU):

$$a[z] = \mathrm{ReLU}[z] = \begin{cases} 0 & \text{if } z < 0 \\ z & \text{if } z \geq 0 \end{cases} \tag{4}$$

The ReLU function returns the input when it is positive and zero otherwise.

Figure 3 shows three distinct functions for y based on ten parameters θ. Each function in Figure 3 is derived from the linear combinations of outputs from hidden units (Equation 5), where each unit represents a linear function $\theta_{\bullet 0} + \theta_{\bullet 1}x$. These linear functions are modified by the Rectified Linear Unit (ReLU) function a[•], which clips values below zero. This clipping introduces non-linear **joints** at points where the lines intersect the x-axis. The modified outputs are then weighted by θ_1, θ_2, and θ_3 respectively, and an offset θ_0 is added to adjust the overall height of the function.

$$\mathbf{y} = \theta_0 + \theta_1 h_1 + \theta_2 h_2 + \theta_3 h_3. \tag{5}$$

The activation pattern of each hidden unit, corresponding to different functions of y, varies. A unit is deemed inactive when clipped by the ReLU function and active otherwise. The slope within each linear region of the function depends on two factors:

the original slope $\theta_{\bullet 1}$ of the active input, and the weights $\theta_{\bullet}$ applied thereafter. With three hidden units, the function can exhibit up to four linear regions, as depicted in Figure 3. However, the slopes of these regions are not entirely independent; the fourth slope is either zero (if all units are inactive) or a combination of the other slopes. This behaviour underscores the complexity and adaptability of such functions in modelling various phenomena. Artificial Neural Networks (ANN) generate predictions $\mathbf{y}$ from inputs $\mathbf{x}$ by segmenting the input domain into a continuous surface of piecewise linear regions. Given a sufficient number of hidden units or neurons, ANNs possess the capability to approximate any continuous function with an arbitrary level of precision.

Neural networks possessing at least one hidden layer are commonly known as multi-layer perceptrons (MLPs). Those with a single hidden layer, as discussed in this chapter, are often termed shallow neural networks. In contrast, networks featuring multiple hidden layers are designated as deep neural networks. When the network's connections create a non-cyclic graph, the structure is identified as a feed-forward network. If each element in one layer is connected to every element in the subsequent layer, as illustrated in the examples of this chapter, the network is described as fully connected. The layers in this network are called dense layers.

For marijuana intoxication detection, MID2021 and MID2023 image datasets are used and convolutional layers are mainly used for processing image data. Convolutional layers process each local image region independently, using parameters shared across the whole image. They use fewer parameters than fully connected layers in ANN, exploit the spatial relationships between nearby pixels, and don't have to re-learn the interpretation of the pixels at every position. A network predominantly consisting of convolutional layers is known as a convolutional neural network or CNN.

Convolutional layers are network layers based on the convolution operation. In 1D, a convolution transforms an input vector x into an output vector z so that each output z_i is a weighted sum of nearby inputs. The same weights are used at every position and are collectively called the convolution kernel or filter. The size of the region over which inputs are combined is termed the kernel size. For a kernel size of

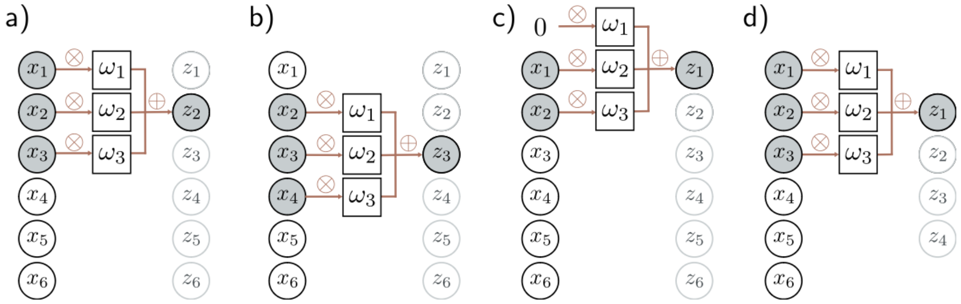


Fig. 4: 1D convolution with kernel size three and stride one. a, b, and c have zero padding, while d has no padding [48].

three, we have:

$$z_i = w_1 x_{i-1} + w_2 x_i + w_3 x_{i+1} \tag{6}$$

where $\mathbf{w} = [w_1, w_2, w_3]^T$ is the kernel.

In Figures 4a and 4b, the outputs z_2 and z_3 are derived as $z_2 = w_1x_1 + w_2x_2 + w_3x_3$ and $z_3 = w_1x_2 + w_2x_3 + w_3x_4$, respectively. Moreover, kernel $\mathbf{w}$ has moved by one unit from receptive field $[x_1, x_2, x_3]^T$ to receptive field $[x_2, x_3, x_4]^T$. The amount with which kernel $\mathbf{w}$ moves from one receptive to another is called stride. Evidently, in Figure 4, if the stride is two, the number of outputs is roughly halved. Figure 4c illustrates that at the z_1 position, the kernel exceeds the first input x_1. This is addressed through zero padding, assuming zero values for inputs outside the range. The end output is processed in a similar manner. Conversely, as shown in Figure 4d, the outputs are calculated exclusively in regions where the kernel is entirely within the input span, resulting in an output size smaller than the input. In Figure 4a, output $[z_1, z_2, z_3, z_4, z_5, z_6]^T$ is one feature map or channel produced by kernel $\mathbf{w} = [w_1, w_2, w_3]^T$. By changing the kernel $\mathbf{w}$, different feature maps are generated by the convolutional layer.

A convolutional layer generates its output by applying a convolution operation on the input, incorporating a bias term (β), and subsequently processing each outcome through an activation function denoted as a[•]. When using a kernel of size three with zero padding and a stride of one, the computation of the ith hidden unit, represented as h_i, is computed as follows:

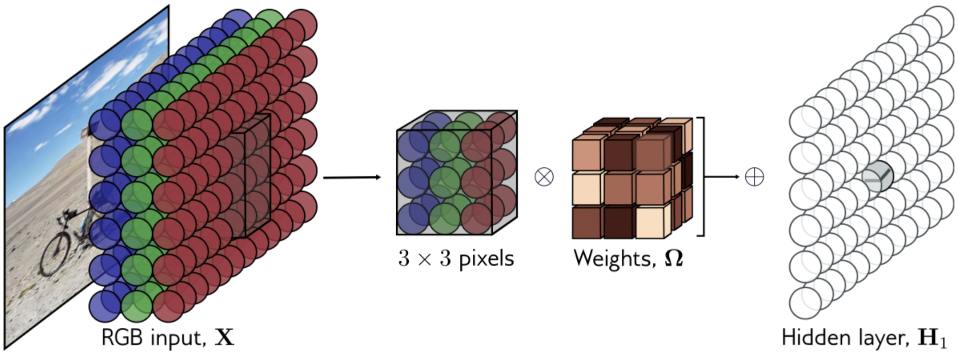


Fig. 5: 2D convolution with kernel size of 3×3 applied to an RGB image [49].

$$h_i = a[\beta + w_1 x_{i-1} + w_2 x_i + w_3 x_{i+1}] \tag{7}$$

So far, convolutional networks have been discussed for processing 1D data. Such networks can be applied to financial time series, audio, and text. However, convolutional networks are more usually applied to 2D image data. Often the input is an RGB image, which is treated as a 2D signal with three channels as shown in 5. Here, a 3×3 kernel would have 3×3×3 weights and be applied to the three input channels at each of the 3×3 positions to create a 2D output that is the same height and width as the input image with zero padding. This process is repeated with different kernel weights to generate multiple output channels and append the results to form a 3D tensor. To reduce the computational load, the memory usage, and the number of parameters, CNN also consists of a pooling layer. The goal of the pooling layer is to subsample the inputs using an aggregation function such as the max or mean.

A convolutional layer simultaneously applies multiple trainable filters to its inputs, making it capable of detecting multiple features anywhere in its inputs. During training the convolutional layer will automatically learn the most useful filters for its task, and the layers above will learn to combine them into more complex patterns. Convolutional layers have been used in StyleGAN3 to generate fine-grained images which we will discuss in the next section. Moreover, the state-of-the-art CNN architecture such as VGG-16, ResNet-50, and Inception-v3 will be discussed in section 2.3 as they have performed well with eye disease detection [11] [37].

2.2 Image Augmentation

Image augmentation is a technique used to artificially increase the size and diversity of image datasets. Traditionally, image augmentation was done by applying various transformations to the original image. The common traditional image augmentation techniques applied to an image are briefly discussed below.

1. **Spatial Transformation:** These transformations include flipping, rotation, and cropping.

2. **Pixel-level Transformations:** These transformations include changing brightness, contrast, and saturation.

3. **Geometric Transformations:** These transformations include affine transformations like translation, shear, and zoom.

4. **Noise Injection:** Some transformations work by adding random noise to images.

Spatial and geometric transformation were applied to MID2021 in [20], which helped proliferate this dataset. However, many deep learning-based techniques could be used to generate new images. The most common one is the General Adversarial Network (GAN), which I will discuss in subsequent sections.

2.2.1 GAN

In the Generative Adversarial Networks (GANs) architecture, the principal generator network synthesizes samples by mapping random noise into the designated output data space. These samples attain plausibility when a discriminator network cannot differentiate between the synthetically generated and real samples. The application of GANs spans diverse data types, encompassing audio, three-dimensional models, textual content, video, and graphical representations. GANs have achieved their most notable successes within image generation, where they can create samples virtually indistinguishable from real pictures.

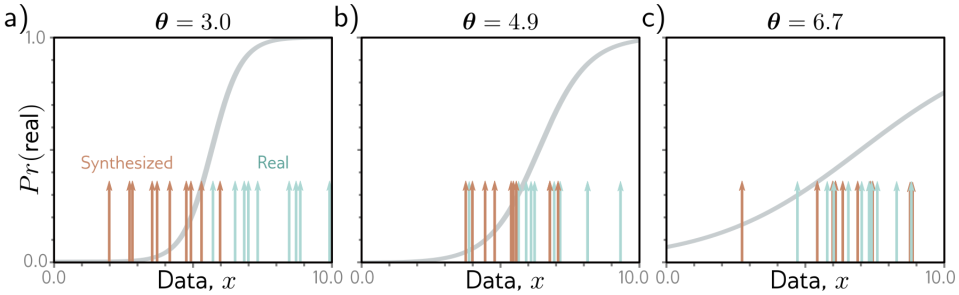


Fig. 6: GAN training on 1D examples [50].

The objective is to produce new samples $\{x^*\}$ that originate from the same distribution as a set of real training data $\{x_i\}$. Each new sample x_j^* is generated by selecting a latent variable z_j from a basic distribution (e.g., a standard normal) and then processing this through a network $g[z_j, \theta]$ with parameters θ, referred to as the generator. The aim during the learning phase is to identify parameters θ such that the samples $\{x^*\}$ resemble the real data $\{x_i\}$.

Generative Adversarial Network (GAN) operates on the principle that the samples should be statistically indistinguishable from the authentic data by employing a second network $f[, \phi]$ with parameters ϕ, known as the discriminator. This network endeavours to classify its input as either a real example of a generated sample. When it becomes challenging for the discriminator to discern between generated and real samples, it suggests that the synthetic examples are convincingly similar to actual ones, marking a successful generative process. On the contrary, if the discriminator identifies differences, it emits a feedback signal that aids in the enhancement and refinement of the generative approach. Figure 6 depicts this concept. The process begins with a training set $\{x_i\}$ of real 1D examples. Each panel i showcases a different batch of ten examples $\{x_i\}_{i=1}^{10}$ (illustrated with cyan arrows). To generate a batch of samples $\{x^*\}$, a straightforward generator is employed:

$$x_j^* = g[z_j, \theta] = z_j + \theta, \tag{8}$$

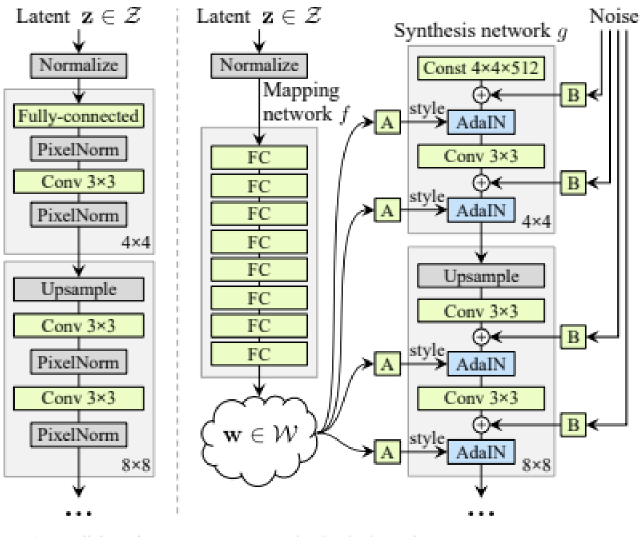
where z_j represents the random noise input, and θ adjusts the position of the generated samples along the x-axis.

Initially set at $\theta = 3.0$, the generated samples (indicated by orange arrows) are positioned to the left of the real examples (cyan arrows). The discriminator is trained to differentiate between the generated samples and the real examples, with the sigmoid curve reflecting the likelihood of a data point being real. As training progresses, adjustments are made to the generator parameters θ to enhance the likelihood of its samples being classified as real. In this context, it involves incrementing θ to shift the samples rightwards, where the sigmoid curve ascends. The training alternates between updating the discriminator and the generator, with Figures **??**b–c demonstrating two iterations of this cycle. Over time, classifying the data becomes increasingly challenging, diminishing the incentive to modify θ as the sigmoid curve levels out. Upon completion, the discriminator and generator become indistinguishable; the discriminator, now operating at chance performance, is set aside. The process concludes with a generator capable of producing convincing samples. Subsequent sub-sections will initially examine the StyleGAN [28] architecture, followed by an exploration of StyleGAN3 [4], which will be employed in the synthesis of images depicting individuals under the influence of marijuana.

2.2.2 StyleGAN

The StyleGAN [28] paper introduces an upgraded iteration of the ProGAN [27], specifically focusing on refining the generator network. The principal innovation embodied by ProGAN [27] centers on its progressive training approach, which commences training by focusing on the generator and discriminator with extremely low-resolution images, such as 4×4 pixels, and progressively introduces higher-resolution layers in subsequent stages.

This method establishes a foundational framework for image synthesis by concentrating on core features prevalent even in low-resolution images. As the training advances, the model incrementally incorporates finer details using different image resolutions. Training on low-resolution images offers the dual benefits of expediency and efficiency, facilitating the training of subsequent higher-resolution layers. Consequently, this progressive training strategy accelerates ProGAN's overall training



(a) Traditional (b) Style-based generator

Fig. 7: ProGAN Generator vs StyleGAN Generator [23].

process. Moreover, it is observed that the lower the layer in the network (and its corresponding resolution), the more it influences coarser image features. These features can be broadly categorized into three types based on resolution:

1. **Coarse**: Resolutions up to 82 pixels, impacting features like pose, general hairstyle, and facial shape.
2. **Middle**: Resolutions ranging from 162 pixels to 322 pixels, affecting finer facial characteristics, hairstyles, and whether eyes are open or closed.
3. **Fine**: Resolutions spanning from 642 pixels to 10242 pixels, governing aspects such as color schemes (eye, hair, and skin) and micro-level features.

Figure 7 shows how the StyleGAN generator incorporates several noteworthy additions compared to ProGAN's generator. The following points discuss the improvements in the StyleGAN3 architecture.

- **Mapping Network**: The mapping network f is a simple feed-forward network that converts the input noise z ∈ Z into a different latent space w ∈ W . This gives the generator the opportunity to disentangle the noisy input vector into distinct factors of variation, which can be easily picked up by the downstream style-generating layers. This task is non-trivial due to the inherent constraint of aligning the input vector with the probability density of the training data. For example, if images of individuals with black hair dominate the dataset, a higher frequency of input values will correspond to this feature. Consequently, the model faces a challenge in disentangling input elements from specific features, a phenomenon called *feature entanglement*. However, this challenge is mitigated by employing an additional neural network, which generates a vector decoupled from the training data distribution, thereby reducing feature correlations. The Mapping Network comprises eight fully connected layers, producing an output denoted as $\boldsymbol{\rho}$ with dimensions identical to those of the input layer, i.e., 512×1.

- **Synthesis Network**: The synthesis network is the generator of the actual image with a given style, as provided by the mapping network. As can be seen from Figure 7, the style vector $\mathbf{w}$ is injected into the synthesis network at different points, each time via a differently densely connected layer A_i , which generates two vectors: a bias vector $y_{b,i}$ and a scaling vector $y_{s,i}$. These vectors define the specific style that should be injected at this point in the network. That is, they tell the synthesis network how to adjust the feature maps to move the generated image in the direction of the specified style. This adjustment is achieved through adaptive instance normalization (AdaIN) layers.

- **Style Modules (Adaptive Instance Normalization):** An AdaIN layer is a type of neural network layer that adjusts the mean and variance of each feature map x_i from mapping network with a reference style bias $y_{b,i}$ and scale $y_{s,i}$, respectively. Both vectors are of length equal to the number of channels output from the preceding convolutional layer in the synthesis network.

  The equation for adaptive instance normalization is as follows:

  $$\mathrm{AdaIN}(x_i, y) = y_{s,i}\frac{x_i - \mu(x_i)}{\sigma(x_i)} + y_{b,i} \tag{9}$$

  where $\mu(x_i)$ and $\sigma(x_i)$ represent the mean and standard deviation of the input feature map x_i.

The adaptive instance normalization layers ensure that the style vectors that are injected into each layer only affect features at that layer, by preventing any style information from leaking through between layers. The authors show that this results in the latent vectors $\mathbf{w}$ being significantly more disentangled than the original $\mathbf{z}$ vectors.

- **Stochastic Variation**: The synthesizer network adds noise (passed through a learned broadcasting layer B as shown in Figure 7) after each convolution to account for stochastic details such as the placement of individual hairs, or the background behind the face. Again, the depth at which the noise is injected affects the coarseness of the impact on the image.

2.2.3 StyleGAN3

The goal of StyleGAN3 [4] is to address the texture sticking problem during the morphing transition (such as changing from one face to another) in StyleGAN [28] and StyleGAN2 [29]. StyleGAN3 seeks to enhance the naturalness of the transition animation. There are different configurations which could help in handling this problem:

- **Configuration A (StyleGAN2):** This is the original setup of the StyleGAN2 architecture, which is used as a reference point for further developments. StyleGAN2 was a notable advancement in generative adversarial networks, producing high-quality images with intricate detail. However, this configuration struggled with maintaining the consistency of textures when an image was subject to spatial transformations such as rotations or scaling. This issue is called texture sticking, where textures maintain their position relative to the pixel grid rather than moving naturally with the object's transformations.

- **Configuration B (Fourier features):** Fourier features are added to the network in this configuration. Fourier transformations convert spatial data into frequency data, representing image characteristics independently of their position in the image space. By using Fourier features, StyleGAN3 aims to learn a more consistent representation of images that is not as sensitive to changes in position, helping to reduce the texture sticking issue.

- **Configuration C (No noise inputs):** StyleGAN2 introduced noise inputs to generate detailed and stochastic variations in textures, enhancing synthetic images' realism. However, this randomness sometimes made textures appear glued to certain coordinates. Configuration C removes these noise inputs, which helps to prevent textures from sticking to fixed locations, thus reducing positional bias in the generated images.

- **Configuration D (Simplified generator):** This configuration simplifies the generator's architecture. The simplification is hypothesized to make the training process more straightforward and the model's behaviour easier to analyze. However, this initial simplification may have slightly impacted the quality of image generation negatively, as indicated by a minor increase in the FID score.

- **Configuration E (Boundaries & upsampling):** This setup aims to improve the generative model's understanding of the broader context within an image by extending the area beyond the immediate pixel neighbourhood, which helps the network maintain the relative positioning of features during transformations. It also incorporates a more sophisticated upsampling method to scale up the resolution of images with fewer artifacts, which could otherwise degrade the image's quality when increasing its size from lower resolutions.

- **Configuration F (Filtered nonlinearities):** This configuration filters the nonlinear activation functions used in the network, such as the rectified linear unit (ReLU). These functions can introduce high-frequency components into the image that are difficult for later layers in the network to process accurately. By filtering these out, Configuration F aims to prevent the creation of textures and patterns that cannot be rendered faithfully, thereby improving the overall quality of the generated images.

- **Configuration G (Non-critical sampling):** The Nyquist rate states that to correctly sample a continuous signal (like sound or an image) and fully reconstruct it later, you need to sample it at a rate that is at least twice the highest frequency present in the signal. This minimum rate is known as the Nyquist rate or Nyquist frequency. In the context of digital images, the concept is similar. The highest frequency would correspond to the finest detail in the image. To capture all the details, you need to have enough pixels per given area to represent the smallest feature. If an image is undersampled, which means sampled below the Nyquist rate for its details, you can get aliasing artifacts—like jagged edges or moiré patterns that weren't present in the original scene.

  Configuration G of StyleGAN3 addresses the principle of non-critical sampling. If the sampling strategy does not respect the Nyquist criterion in image generation, it could lead to aliasing, where generated textures show patterns or distortions that make them look unrealistic. By ensuring that the model samples at or above the required rate for the image details it generates, the resulting images are free from such artifacts and look more natural, especially during transformations like rotation or scaling.

- **Configuration H (Transformed Fourier features):** Building on Configuration B, this setup transforms the Fourier features to better align with the network's internal operations. This transformation enables the network to manage spatial changes more effectively, addressing the sticking problem where certain features might appear static or unnaturally tied to specific pixel coordinates during transformations.

- **Configuration T (StyleGAN3-T):** T stands for translation equivariance, which means that the network ensures consistent behaviour of textures and features when an object moves across the image plane. This configuration is fine-tuned to make the model more sensitive to translations, so textures and details in generated images move naturally and in coordination with the objects they belong to.

- **Configuration R (StyleGAN3-R):** *R* stands for rotation equivariance, optimized for maintaining texture fidelity during rotations. This configuration signifies a significant advancement in GAN architecture by effectively managing the complex task of rotating images or features within images without losing texture quality. Due to this, model succeeds in producing images where features and textures appear natural and realistic, even after rotation.

Configuration	FID ↓
A StyleGAN2	5.14
B + Fourier features	4.79
C + No noise inputs	4.54
D + Simplified generator	5.21
E + Boundaries & upsampling	6.02
F + Filtered nonlinearities	6.35
G + Non-critical sampling	4.78
H + Transformed Fourier features	4.64
T + Flexible layers (StyleGAN3-T)	4.62
R + Rotation equiv. (StyleGAN3-R)	4.50

Table 1: FID for various configurations of StyleGAN2 and StyleGAN3.

The table 1 presents performance metrics for different configurations of StyleGAN2 and StyleGAN3, illustrating the evolution and enhancement of these generative models in terms of image quality and geometric transformation handling. StyleGAN2 serves as the base model, with a decent Fréchet Inception Distance (FID) score, measuring the distance between the feature vectors of real and generated images; a lower score indicates better image quality. The subsequent configurations from B to H incrementally introduce improvements. Fourier features are added to help the network learn translation-invariant representations, and removing noise inputs mitigates positional bias. Simplifying the generator, enhancing boundaries and upsampling methods, applying filtered nonlinearities, and adjusting the sampling strategy all aim to improve the model's equivariance to translation and rotation. Each modification improves FID, EQ-T, and EQ-R scores, reflecting the trade-offs between image quality and transformation handling. Most of these developments are found in StyleGAN3's T and R configurations. The *T* variant is tuned for translation equivariance, achieving an impressive balance between image quality and the ability to handle translations. The *R* variant is optimized for rotation equivariance, showcasing an extraordinary ability to maintain image realism when the generated images are rotated. This is evidenced by its high EQ-R score, the best of all configurations, demonstrating StyleGAN3-R's advanced capability to produce consistent and high-fidelity images across various transformations. This next section will discuss relevant knowledge on three CNN architectures [11] [37] for this thesis.

2.3 CNN Architectures

2.3.1 VGG-16

VGG-16 [19] [52] is a widely used algorithm for both object detection and image classification, capable of categorizing images into 1,000 different classes. It is particularly user-friendly for transfer learning applications. The designers of VGG-16 enhanced its performance by opting for a deep architecture featuring small 3x3 convolutional filters. This design choice led to substantial improvements over previous models. With a depth ranging from 16 to 19 layers, VGG-16 has around 138 million trainable parameters (weights between layers). Figure 8 shows the architecture of VGG-16 architecture and explanation of VGG-16 architecture is given below.

- VGG-16 takes input tensor size as 224 × 224 with 3 RGB channel.
- The 16 in VGG-16 refers to sixteen layers with weights or layers that learn the pattern from the input data. In VGG-16, there are three kinds of layers: convolutional, max pooling, and dense layers. Out of the three layers, only the Max pooling layer is not learnable, as its purpose is to reduce the spatial

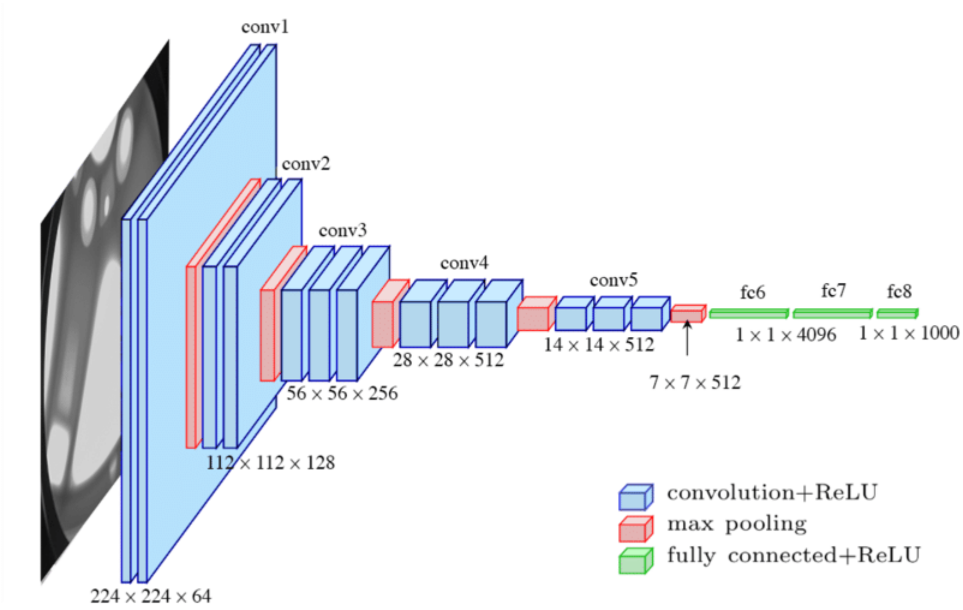


Fig. 8: VGG-16 Architecture [19].

dimension of input feature maps. However, it has only sixteen weight layers, i.e., thirteen Convolutional layer and three fully connected layers, both with ReLU as the activation function. The convolution and max pool layers are consistently arranged throughout the whole architecture.

- Most unique thing about VGG-16 is that instead of having a large number of hyper-parameters, they focused on having convolution layers of 3×3 filter with stride 1 and always used the same padding and maxpool layer of 2×2 filter of stride 2.

- In Figure 8, the VGG-16 architecture is depicted with five distinct convolutional layers: conv1, conv2, conv3, conv4, and conv5. The conv1 layer comprises 64 filters, conv2 has 128 filters, conv3 is equipped with 256 filters, and conv4 and conv5 contain 512 filters each.

- In Figure 8, the VGG-16 architecture is also shown with three fully connected layer: fc6, fc7 and fc8. The first two layers, fc6 and fc7, each consist of 4,096 neurons and are responsible for processing high-level feature representations obtained from preceding convolutional layers. These neurons add a non-linear component to the network by using the Rectified Linear Unit (ReLU) activation function. The output layer is the terminal densely connected layer (fc8), which has 1,000 neurons in a layout designed specifically for the ImageNet classification task. The softmax activation function is used in this output layer to convert the neural responses into probability scores for each class. The class that has the highest probability becomes the prediction of the network.

2.3.2 Inception-v3

Inception V3 [56] [55], a convolutional neural network (CNN) architecture, was developed by researchers at Google and represents a significant milestone in the evolution of deep learning models for image classification. Inception V3, or GoogLeNet V3, was introduced to address the growing demand for more efficient and accurate image recognition algorithms. Before discussing the architecture of Inception-v3, we will briefly discuss the Inception-v1 architecture.

Inception V1: In the Inception-v1 architecture, often referenced as GoogLeNet, the introduction of the Inception module marked a significant advancement in convolutional neural network design. This module was ingeniously architected to concurrently capture multi-scale spatial information using an array of convolutional filters of varying dimensions. The primary variant or naive version of the Inception module combines four parallel operations: a direct 1×1 convolution, 3×3 convolution, 5×5 convolution, and a 3×3 max pooling. The outputs of these parallel components are concatenated to constitute the unified output of the module. A secondary, more expansive variant of this module consisted of a direct 1×1 convolution, a dimensionality-reducing 1×1 convolution followed by a 3×3 convolution, a similar sequence leading to a 5×5 convolution, and a 3×3 max pooling succeeded by a 1×1 convolution. This more expansive variant of this module amplifies the 1×1, 3×3, and 5×5 convolutions, thereby widening the module's scope without substantially increasing computational burden. This multi-scale approach empowers the network to autonomously discern the optimal filter dimensions for each layer autonomously, fostering a richer and more nuanced representation of input data.

Inception V3 uses inception modules, which act as lenses for image analysis. Some modules focus on fine details, while others examine larger shapes. By leveraging these diverse lenses, Inception V3 gains a comprehensive understanding of intricate details and the overall context within an image. Introduced by Christian Szegedy and colleagues at Google, it comprises 42 layers and has a lower error rate than its predecessors. Let's delve into the optimizations that make the Inception V3 model superior.

The key modifications made to the Inception V3 model are as follows:

1. **Factorization into Smaller Convolutions:** To enhance the dimension reduction aspect, larger convolutions in the model were factorized into smaller convolutions. Figure 9 shows two similar networks concatenating filters of different sizes. The only difference between the two diagrams is that the 5×5 convolutional layer on the left side of the image is replaced with two 3×3 convolutional layers on the right side of the image, which significantly reduces computational costs.

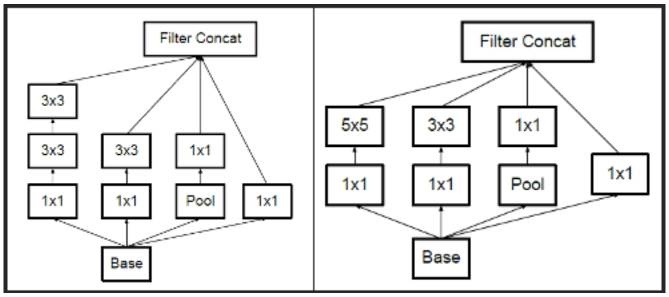


Fig. 9: Factorization Example [56].

2. **Spatial Factorization into Asymmetric Convolutions:** Asymmetric convolutions in the form of n×1 and 1×n were introduced as an efficient alternative to factorize larger convolutions of n×n. The 3×3 convolutions is replaced with 1×3 followed by 3×1 convolutions, maintaining the same receptive field as a 3×3 convolution while reducing computational demands.

3. **Utility of Auxiliary Classifiers:** In a typical deep neural network, the loss is calculated only at the output layer. However, in architectures like Inception-V3, additional classifiers (often called auxiliary classifiers) are attached to intermediate layers of the network. These auxiliary classifiers also contribute to the total loss, but usually with less weight than the primary classifier at the output. Auxiliary classifiers aid deep neural network convergence and mitigate the vanishing gradient problem. While these classifiers did not show immediate improvements during early training stages, they enhanced accuracy towards the end of training, acting as regularizers in the Inception V3 architecture.

4. **Efficient Grid Size Reduction:** Traditionally, researchers used max pooling and average pooling to reduce the grid size of feature maps. However, in the inception V3 model, they opted to efficiently reduce the grid size by expanding the activation dimension of the network filters. For instance, if we start with a d×d grid and k filters, the reduction process results in a d/2 × d/2 grid with 2k filters. This is achieved by employing two parallel blocks of convolution and pooling.

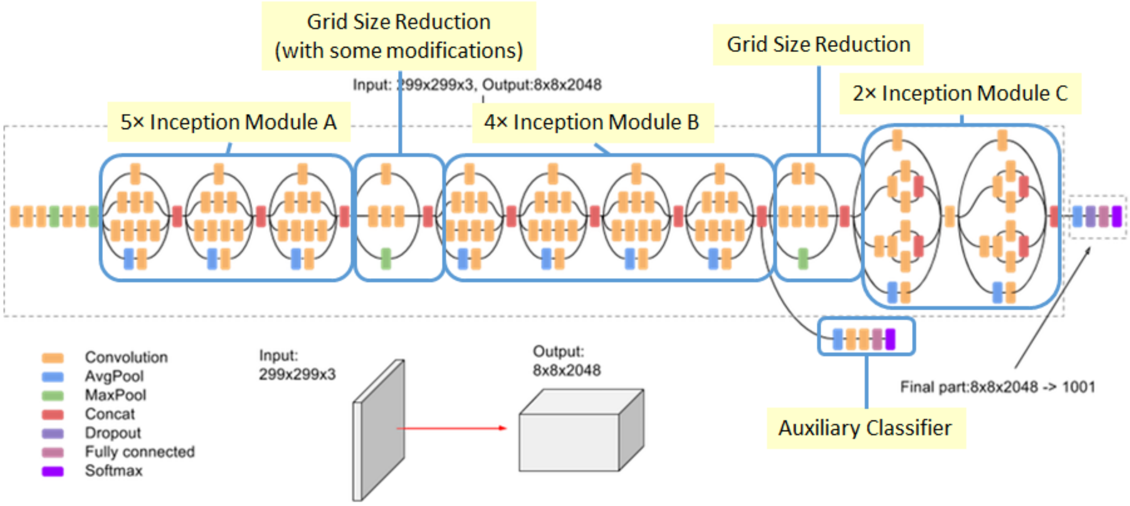


Fig. 10: Inception V3 Architecture [56].

After implementing these optimizations, the Inception V3 model comprises 42 layers, making it slightly deeper than its predecessors. However, its efficiency and performance improvements are remarkable. The below points outline the architecture of the Inception V3 model as shown in Figure 10:

1. **Convolutional Layers:** The table begins with two convolutional layers. The first layer applies a 3x3 convolutional operation with a stride of 2 to the input image of size $299 \times 299 \times 3$, producing 32 feature maps with dimensions $149 \times 149 \times 32$. The second layer is similar but uses a stride of 1, retaining the spatial dimensions.
2. **Padded Convolution:** A third convolutional layer is introduced with a 3x3 kernel and a stride of 1. It uses padding to ensure that the spatial dimensions remain at $147 \times 147 \times 32$.
3. **Pooling Layer:** Following the padded convolution, a pooling layer with a 3x3 window and a stride of 2 is applied. This operation results in 64 feature maps with dimensions $73 \times 73 \times 64$.
4. **Inception Module A:** This is where the Inception architecture comes into play. Three consecutive Inception blocks are applied. Each block contains multiple parallel convolutional operations with different kernel sizes and depths, allowing the model to capture a wide range of features. This module results in feature maps of size $35 \times 35 \times 288$.
5. **Inception Module B:** Module B expands on the concept of Module A and consists of five Inception blocks. These blocks further increase the depth and complexity of feature extraction, resulting in feature maps of size $17 \times 17 \times 768$.
6. **Inception Module C:** Module C follows Module B and includes two Inception blocks. It continues to capture intricate features and reduces the spatial dimensions to $8 \times 8 \times 1280$.
7. **Global Average Pooling:** After the Inception modules, global average pooling is applied to the feature maps. This operation reduces the spatial dimensions

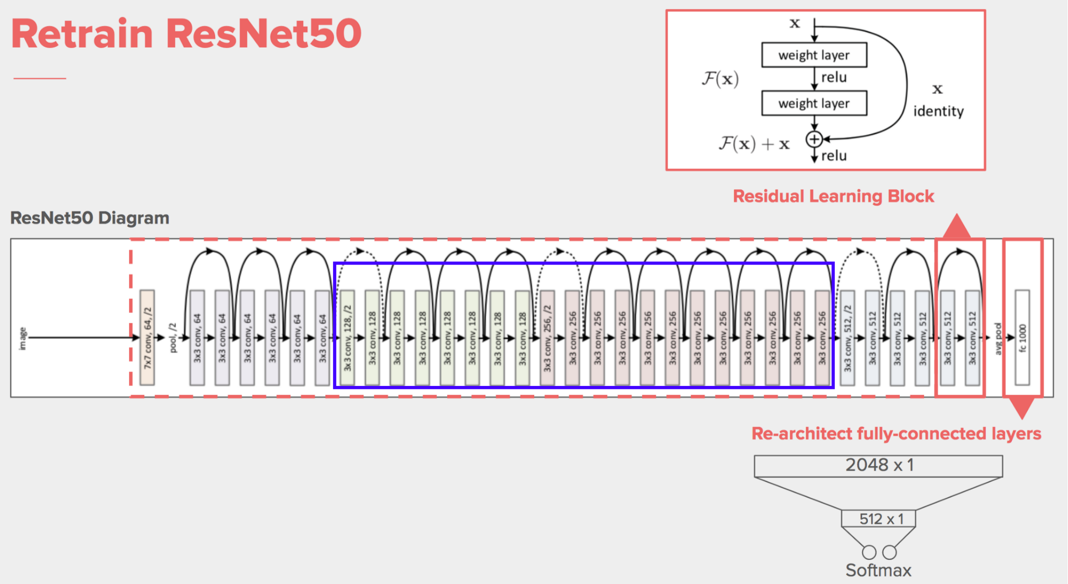


Fig. 11: ResNet-50 Architecture [16].

to $8 \times 8 \times 2048$ by taking the mean value of each feature map, resulting in a compact representation.

8. **Fully Connected Layers:** The flattened output from global average pooling serves as input to fully connected layers, further processing the feature information. The penultimate layer has 2048 neurons, and the final layer, is referred to as Linear (Logits), consists of 2048 neurons.

9. **Softmax Classifier:** The last layer applies the softmax activation function to produce class probabilities for 1000 categories, as this model is designed for a 1000-class image classification task.

2.3.3 ResNet-50

ResNet-50 [22] [16] is a revolutionary neural network model primarily designed for image classification, part of the ResNet family, which also addresses the vanishing gradient problem in deep neural networks. The model is widely recognized for its residual learning framework, which facilitates the training of extremely deep networks.

Figure 11 shows the architecture of ResNet-50 and the module named Residual Learning Block or skip connection in ResNet-50. The architecture of ResNet-50 and its components are explained below.

- ResNet-50 is named for its 50 layers, out of 48 are Convolutional layers, 1 is a Max Pooling layer, and 1 is an Average Pooling layer.
- The input to ResNet-50 is a 224x224 image with 3 RGB channels.
- A distinctive feature of ResNet-50 is using skip connections, or shortcuts, to jump over some layers, which helps prevent the vanishing gradient problem in deep networks.
- ResNet-50 is structured in a way that it has a series of residual blocks containing three layers each. Each residual block has a skip connection that bypasses the two convolutional layers.
- The architecture utilizes bottleneck design, where three layers are stacked together, and dimensionality is reduced and then increased within the residual block to improve efficiency.
- The last layer of ResNet-50 is a fully connected layer with 1,000 neurons (one for each class) followed by a softmax activation function to output the class probabilities.

ResNet-50, with its innovative architecture, remains a popular choice for various computer vision tasks, including object detection and image classification. It is particularly effective for transfer learning applications due to its deep architecture and residual learning framework.

In this chapter, we delve deeply into the foundational concepts and technical background information related to this thesis's primary discussion topics. Chapter 3 further expands upon this by presenting a detailed literature review examining the various techniques employed in Marijuana intoxication detection, as initially introduced in Chapter 1. This literature review traces the evolution of these techniques and critiques their effectiveness and applicability in real-world scenarios.

CHAPTER 3

Literature Review

Cannabis, commonly called as marijuana, has seen its usage rates rise globally for medicinal and recreational purposes, a trend elaborated upon in Chapter 1. As its prevalence continues to climb, there is a growing emphasis on methods for detecting its usage and understanding its effects, especially in scenarios like roadside and occupational drug screenings. Conventional, non-machine learning approaches for detecting cannabis use have demonstrated a range of limitations. Section 3.1 to Section 3.4 discuss the literature on non-machine learning-based marijuana intoxication detection techniques. In section 3.5, we will discuss the literature on how deep-learning image augmentation techniques helps in improving model performance and how different CNN architectures help in improving performance of model which relies on eyes as features.

3.1 Field Sobriety Tests and Drug Recognition Experts (DREs)

Field sobriety tests (FSTs) are often the first line of assessment when there is suspicion of drug-induced impairment, especially in roadside situations. FSTs typically involve physical and cognitive tasks designed to gauge potential impairment. However, the correlation between FST results and blood THC concentration is not straightforward, leading to potential inconsistencies and inaccuracies [53].

To improve upon the reliability of FSTs, law enforcement agencies have adopted the use of Drug Recognition Experts (DREs). DREs are officers who have undergone extensive training to recognize drug-induced impairment. Their evaluations follow a standardized 12-step process that includes examining the suspect's medical history, vital signs, eye movements, and performance on psychophysical tests [43].

Despite these advancements, both FSTs and DRE evaluations have significant limitations. The subjectivity involved in these assessments can lead to inconsistent results. For instance, two officers might interpret the results of the same FST differently. Additionally, the DRE's 12-step evaluation process, while standardized, is still reliant on the DRE's judgment and interpretation [43] and, while FSTs and DRE evaluations can provide some indication of cannabis intoxication, they are not wholly reliable or accurate [43].

3.2 DRUID Project

The DRUID (Driving under the Influence of Drugs, Alcohol and Medicines) project is a large-scale European initiative that aims to combat drug-impaired driving by providing significant resources and guidelines for DRE evaluations [43]. The project involved numerous research institutions, universities, and public interest groups that worked together to gather and analyze data on substance use and driving.

One of the key outcomes of the DRUID project was the standardization of DRE evaluations. The project provided extensive resources and guidelines for DREs, which have improved the consistency and reliability of DRE evaluations across Europe [43].

Despite these advancements, DRE evaluations, even when guided by the DRUID project's resources, are not without their limitations. The evaluations are still dependent on the DRE's judgment, and there remains a level of subjectivity in the process [43]. Additionally, DRE evaluations require extensive training and are only sometimes available when needed.

3.3 Oral Fluid Testing

Oral fluid testing is a non-invasive method used to detect cannabis use. This method analyzes an individual's oral fluid (saliva) for THC and other cannabinoids. A systematic review on the correlation between oral fluid and blood THC concentration found that while there is a correlation, it is not consistent or strong enough to allow for the precise estimation of blood THC concentration based on oral fluid tests [51].

Furthermore, oral fluid testing can sometimes lead to false positives due to THC in the mouth immediately after smoking or consuming cannabis, without the person being impaired [26]. This indicates that while oral fluid testing can provide some indication of cannabis use, it alone may not be sufficient to determine impairment.

3.4 Breath Testing

Breath testing is a relatively new method for detecting cannabis use. This method analyzes an individual's breath for certain volatile organic compounds (VOCs) specific to cannabis use. A pilot study developed a comprehensive breath test confirming recent cannabis use within the impairment window [17].

However, this method is still in the early stages of development, and further studies are needed to validate its effectiveness [17]. Additionally, the breath test was designed for inhaled cannabis, and it is unclear whether the breath test would be effective in detecting cannabis use through other routes of administration [17].

3.5 Marijuana intoxication detection using Machine Learning

As discussed in section 1.4.5, Gadhiya [20] thesis only focuses on machine learning based approach in detecting marijuana intoxication by creating MID2021. MID2021 [20] was sourced from YouTube videos where individuals have recorded themselves smoking marijuana. Clippings of individuals' faces were taken from these videos and they gathered approximately 15-20 images per video. These images showed both the states, sober and then intoxicated, of the same person at different angles. Traditional augmentation were applied to collected images which increased the count of dataset from 600 to 2750. However, deep learning-based image augmentation have gained attention across various sectors, encompassing medical imaging, dermatological diagnosis, character animation, style transference, and image identification.

In the field of medical imaging, Chlap et al. [15] conducted a systematic review of data augmentation techniques used in deep learning applications. They highlighted the importance of data augmentation in addressing challenges such as limited data availability and class imbalance. Benalcazar et al. [9] proposed a novel data augmentation technique to enhance iris recognition accuracy, particularly under varying pupil dilation conditions. The methodology involves artificially transforming iris images to represent different dilation levels, significantly expanding the training dataset's diversity. For the experiments, the team utilized a subset of a larger dataset, comprising 2,400 images from 120 subjects, capturing a wide range of natural pupil dilations. They implemented their unique Artificial Dilation Data Augmentation method, generating 19 artificial dilations for each image, resulting in up to 38,400 images per training epoch. This approach was compared against a Normal data augmentation (Traditional data augmentation) scenario, employing standard augmentations like scaling, flipping, and brightness adjustment. The experiments were conducted across seven different neural network models to ensure consistency and reliability in the results. Each network was trained from scratch under both scenarios, maintaining identical hyper-parameters to ensure comparability. The results were compelling, indicating a substantial improvement in segmentation accuracy up to 15% for images with high pupil dilation using the Artificial Dilation DA method.

Once the MID2021 was assembled and augmented, Gadhiya [20] conducted 6 different experiments by using SVM, Random Forests, Decision Trees, and MobileNet to discern marijuana intoxication in individuals. Moreover, 3-fold cross-validation approach was applied to the SVM model and MobileNet model. The performance metrics for various models in [25] revealed that the MobileNet outperformed others with a precision of 80%, recall of 83%, an f1-score of 82%, and an accuracy of 82%. Overall, the MobileNet (CNN) emerged as the most efficient model. MobileNet neural network architecture begins with two convolutional layers. The first layer has a kernel size of 3×3, a stride of 1, and padding of 1. The second convolutional layer maintains these parameters. A max-pooling layer follows, with a kernel size of 2 and a stride of 2, reducing the spatial dimensions of the feature maps by half without padding. The network also includes two fully-connected layers outputting two classes, sober or intoxicated, for binary classification. A dropout layer with a 0.5 rate is utilized between two fully connected layers to mitigate overfitting. MobileNet is efficient model for mobile and embedded vision applications. However, MobileNet was neither adequately deep nor sufficiently broad to proficiently extract discernible features from images of marijuana intoxication. Since bloodshot or red eyes are considered features for intoxication detection, it's essential to explore suitable architectures that excel in eye-related tasks. In the domain of eye disease prediction, machine learning algorithms have gained widespread use in developing accurate diagnostic models. Notably, two recent studies, Marouf et al. [37] and Bitto et al. [11], have introduced efficient approaches for predicting eye diseases using machine learning.

Marouf et al. [37] proposed a comprehensive machine learning (ML) approach for the early diagnosis of five common eye diseases: Cataracts, Acute Angle-Closure Glaucoma, Primary Congenital Glaucoma, Exophthalmos, and Ocular Hypertension. Leveraging a benchmark dataset annotated by practicing ophthalmologists, the study employed nine classic ML algorithms, including Support Vector Machine (SVM) and Logistic Regression (LR), achieving remarkable accuracies—up to 99.11% with SVM. To enhance model robustness, the research applied multiple feature selection methods and stratified 10-fold cross-validation. Traditional performance metrics, including accuracy, precision, sensitivity, and F1-Score, were utilized for evaluation. The study identified critical disease symptoms through feature selection, offering valuable insights into diagnosis. The paper also envisions future directions, such as multivariate analysis for deeper symptom insights and the integration of explainable AI techniques. Overall, the study represents a promising step towards affordable and efficient eye disease diagnosis, especially in settings where expert medical evaluation is scarce or expensive.

Bitto et al. [11] directed their focus towards detecting common eye diseases using a transfer learning approach with CNN. The authors harnessed pre-trained CNN models and fine-tuned them on a dataset comprising images of eye diseases. This study harnesses the potential of Convolutional Neural Networks (CNNs) for diagnosing eye diseases, with a specific focus on distinguishing between normal eyes, eyes with conjunctivitis, and eyes with cataracts. Three popular deep learning architectures—VGG-16, ResNet-50, and Inception-v3—were employed for this task, utilizing a robust dataset of photographs collected from the internet. The research leverages Transfer Learning (TL) and deep feature extraction techniques to enhance the model's performance. Among these models, Inception-v3 stands out with the highest accuracy of 97.08% and the quickest detection time of 485 seconds. ResNet-50 follows with an accuracy of 95.68% but takes 1090 seconds, while VGG-16 also delivers commendable performance with 95.48% accuracy, albeit with the longest detection time of 2510 seconds. This study aims to extend their work by exploring other CNN architectures and data augmentation techniques. It underscores the potential of using smartphone cameras for real-time eye disease detection, which could revolutionize quick and accessible diagnoses. The next chapter will discuss the research objectives and motivation behind them based on the literature discussed for machine learning-based approach to marijuana intoxication detection [20].

CHAPTER 4

Motivation and Research Objective

4.1 Motivation

Currently, there is a lack of an effective detection system for identifying marijuana intoxication without the need for samples like blood, breath, or plasma. Traditional methods often require physical contact or touch, which can be invasive and impractical in many scenarios.

The advancement of artificial intelligence (AI) and machine learning models provides an opportunity to address this challenge. These technologies can be harnessed to recognize signs of marijuana intoxication based on eye movements, facial expressions, and other visual cues present in images and videos.

The motivation behind developing a machine learning-based approach for detecting marijuana intoxication using visual data is to enable early remote detection. Such a system has the potential to play a crucial role in preventing and reducing motor vehicle crashes and occupational injuries in workplace settings.

By leveraging AI and machine learning, this thesis aims to improve the performance of work by Gadhiya [20] for identifying marijuana intoxication, enhancing safety, and mitigating risks associated with impaired individuals in various environments.

4.2 Research Objective

Chapter 3 discussed that Gadhiya [20] work only focuses on a machine learning-based approach to detect marijuana intoxication. The key areas which were novel in [20] were data collection from the internet, traditional augmentation, to increase the dataset and a machine learning approach to identify the intoxicated person.

The advent of deep learning-based image augmentation techniques has broadened the horizons for artificial data generation, including creating images of non-existent individuals. These techniques enhance model performance and add diversity and variability to training datasets. The release of StyleGAN3 marks a significant milestone, offering advanced image manipulation and style-based transformations. Such advancements have been instrumental in bolstering the robustness and effectiveness of deep learning models across various applications.

The study by Bitto and Mahmud et al. [11] focuses on using pre-trained CNN models, specifically VGG-16, ResNet-50, and Inception-v3, for diagnosing common eye diseases like conjunctivitis and cataracts. Employing a transfer learning approach on a robust dataset, the study finds that Inception-v3 has the highest accuracy at 97.08% and the fastest detection time of 485 seconds. While, ResNet-50 and VGG-16 also show high accuracy but take longer to detect. The application of VGG-16, ResNet-50, and Inception-v3 models in [11] shows that these models could work well for images of marijuana intoxicated individuals.

The main objectives of this thesis encompass two critical aspects of marijuana intoxication detection in [20]:

1. **Expansion of Bloodshot Eyes Dataset:** This thesis explores the feasibility of enlarging the Bloodshot Eyes Dataset. This expansion will be orchestrated by using image augmentation and advanced generative image techniques such as StyleGAN3. We aim to enhance the quality and quantity of images available in the dataset and diversify the images containing bloodshot eyes, a crucial indicator of marijuana intoxication. A more extensive and varied dataset can improve model training and lead to more accurate detection.

2. **Enhancement of Marijuana Intoxication Detection Models:** This thesis seeks to elevate the performance of existing marijuana intoxication detection models by employing CNN models, which have deep or sufficiently broad networks to extract discernible features from images of marijuana intoxication proficiently. To achieve this, state-of-the-art deep learning models, including VGG-16, ResNet-50, and Inception-v3, will be integrated into the detection framework. Leveraging these advanced architectures can enhance the model's performance employed for marijuana intoxication detection by improving the knowledge learned during its training and will allow it to become more robust in dealing with different variants of images of bloodshot eyes.

CHAPTER 5

Expansion of Bloodshot Eyes Dataset

This chapter delves into our first research objective: Expanding the bloodshot eyes image dataset by using deep learning-based image augmentation, StyleGAN3, on the MID2021 and CelebA [32] to create a new dataset, MID2023. The performance of the dataset is assessed by computing the FID (Fréchet Inception Distance) score. A lower FID score indicates that the images generated are of relevant use case and appear in line with our original set of images. Section 5.1 discusses in detail our experiments, which include details of the dataset and the evaluation metrics utilized. Section 5.2 details the pipeline we implemented to generate MID2023. Section 5.3 lists and discusses the results obtained from the experiments.

5.1 Dataset and Evaluation Metric

Different sets of experiments were conducted on StyleGAN3-R variant with MID2021 as StyleGAN3-R has a low FID score, signifying that the image is less distorted from the original image and follows the same structure as the original image. MID2021 was first pre-processed to follow the convention of images as suggested in StyleGAN3. The pre-processed MID2021 was first trained on two nodes of Sharcnet Graham P100 Pascal GPUs [1] with 50 GB RAM for three days. The following subsection will discuss both the MID2021 and CelebA dataset, utilized in training the StyleGAN3-R variant, and the evaluation methodolgy applied to StyleGAN3-R for the experiments.

5.1.1 Dataset

MID2021: The dataset was created [20] by gathering images of marijuana-intoxicated persons from Google and YouTube searches. It contained 2,750 images, out of which half the images were of sober eyes, and the other half had images of marijuana-intoxicated eyes. Classical image augmentation techniques, such as image rotation, random cropping, flipping, and adjustments to image contrast, were applied to these images. This augmentation helped expand the dataset to include 1375 images of intoxicated individuals and equivalent images depicting sober individuals. Table 2 shows the number of original and intoxicated images in the MID2021. We can infer from the table 2 that two classes, stoned and sober, have equal images.

Number of Original Sober Images	300
Number of Original Intoxicated Images	300
Number of Original Images	600
Number of Augmented Sober Images	1375
Number of Augmented Intoxicated Images	1375
Number of Augmented Images	2750

Table 2: MID2021 Summary

CelebA Dataset: CelebFaces [32]Attributes Dataset (CelebA) is a large-scale face attributes dataset with more than 200K celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter. CelebA has large diversities, large quantities, and rich annotations, including

- 10,177 number of identities,
- 202,599 number of face images, and
- 5 landmark locations, 40 binary attributes annotations per image

For the conducted experiments, a total of 600 images were utilized for 600 original images in MID2021 [20].

5.1.2 Evaluation Metric

Evaluation metrics are quantitative measures that are used to assess the performance of a machine learning model. To evaluate the performance of StyleGAN3, we employed the Fréchet Inception Distance (FID) score as an evaluation metric.

The Fréchet Inception Distance (FID) score is a measure used to evaluate the quality of images generated by generative models, such as StyleGAN3. It calculates the distance between the feature vectors of real and generated images. This equation provides a metric that compares the statistical properties (mean and covariance) of the features of real and generated images. A lower FID score indicates that the two sets of images are more similar; hence, it is often used to indicate better generative model performance.

The FID score is computed as:

$$\mathrm{FID}(x, g) = ||\mu_x - \mu_g||^2 + \mathrm{Tr}(\Sigma_x + \Sigma_g - 2(\Sigma_x \Sigma_g)^{0.5})$$

where μ_x and μ_g are the means of the real and generated feature vectors, respectively, Σ_x and Σ_g are the covariances of the real and generated feature vectors, respectively, and, Tr denotes the trace of a matrix.

5.2 Flow for Dataset Creation

This section will discuss the flow used in this thesis to create new images from StyleGAN3. Images generated by StyleGAN3 and the original images in MID2021 are used to create a new dataset, MID2023. Firstly, we will pre-process original images in MID2021, and selected images from CelebA to generate original MID2021 faces. Following this, we will create new images by training StyleGAN3-R with pre-processed images from MID2021 and CelebA.

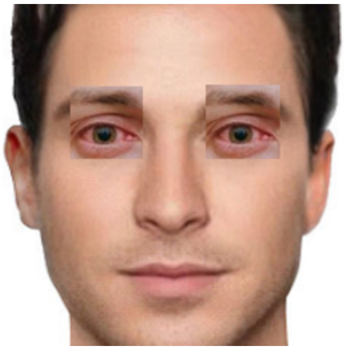


Fig. 12: Pre-processed image from MID2021 and CelebA.

5.2.1 Dataset Pre-processing

We are going to utilize bloodshot eyes or red eyes as our choice of facial feature in this work to detect marijuana consumption. We first detected and cropped eyes for CelebA selected images using the dlib library. Each original eye images in MID2021 is resized to the dimensions of the cropped CelebA selected eye images. Then, the CelebA eye images are replaced with re-scaled MID2021 original eye images in CelebA selected images. This way, we can generate the whole face of a marijuana-intoxicated person. Figure 12 shows a marijuana intoxicated face generated after pre-processing datasets. The following points explain the pre-processing steps performed on images before using them in StyleGAN3, which will be discussed in section 5.2.

- As discussed in section 2.1, an entire image $\mathbf{I}$ can be thought of as a matrix comprised of three channels $\mathbf{R}$, $\mathbf{G}$, and $\mathbf{B}$ each of dimensions $M \times N$ stacked together. Section 1.4.5 discussed that Dlib makes eye detection convenient by finding the left eye at detection points 37-42 and the right eye at detection points 43-48, which can be seen in Figure 1. Using these landmark points, the dlib library can give an accurate image of an eye. $\boldsymbol{E_c}$, with three 2D matrices as $\boldsymbol{R_c}$, $\boldsymbol{G_c}$ and $\boldsymbol{B_c}$ of dimensions $M_c \times N_c$, where, $c \in$ CelebA selected images.

- Using the same process, we can get images of eyes $\boldsymbol{E_o}$, with three 2D matrices as $\boldsymbol{R_o}$, $\boldsymbol{G_o}$ and $\boldsymbol{B_o}$ of dimensions $M_o \times N_o$ from image of original eyes, $\boldsymbol{I_o}$, where $o \in$ MID2021 original eyes. Depending upon the size difference between $\boldsymbol{E_c}$ and $\boldsymbol{E_o}$, $\boldsymbol{E_o}$ is up-scaled or down-scaled to $\boldsymbol{E_o'}$ using OpenCV library such that $\boldsymbol{M_o} = \boldsymbol{M_c}$ and $\boldsymbol{N_o} = \boldsymbol{N_c}$. Then, the CelebA eye image, $\boldsymbol{E_c}$ is replaced with re-scaled original eye in MID2021, $\boldsymbol{E_o'}$ in sober image, $\boldsymbol{I_c}$.

After these steps, MID2021 is pre-processed to generate full faces with marijuana-intoxicated eyes that can be used in training StyelGAN3.

5.2.2 Augmentation of Images via StyleGAN3

To make our model more robust, we employ deep learning-based image augmentation techniques to have more variations of the existing data in our dataset. For this purpose, we employ StyleGAN3 [4], known for its superior image generation capabilities. We train StyleGAN3 on the pre-processed images derived from the pre-processing step. Our choice of StlyeGAN3 variant involves using StlyeGAN3-R for our experimentation as this variant outperforms other variants of stlyeGAN3 when compared with their respective FID scores as shown in Table 1. The following points explain the steps involved in augmenting images using StyleGAN3:

- **Pre-processing in StyleGAN3**: For StyleGAN3, there are specific image requirements during pre-processing. Firstly, the output images must be square-shaped, with their dimensions being a power of two. Additionally, there's flexibility in setting precise dimensions for these images via the resolution option. Furthermore, the system provides transformation tools specifically for cropping purposes. Users can opt between a center crop or a broader center crop, both of which should be paired with the specified resolution to ensure optimal results. These guidelines ensure images are appropriately formatted for the subsequent training processes. After this step, dataset metadata and labels are stored in a JSON file.

- **StyleGAN3 Training**: Training StyleGAN3 involves a detailed interplay of various hyper-parameters, each serving a unique role to ensure the model trains effectively and efficiently. The *snap* parameter determines the time intervals at which the network's progress snapshots are saved, essential for monitoring progress and potentially resuming interrupted training sessions. For our case, *snap* is 100 as the model's progress will be saved every 100 kimg (thousands of images). StyleGAN3 has a long training time spanning across days, and due to computational limitations, we pause and resume the model's training using the *resume* parameter of StyleGAN3. The *resume* option allows training to continue from a previously saved state, which is invaluable for extended training sessions. However, this re-training has caused some spikes, which we will discuss in section 5.3. The cfg argument denotes the specific configuration employed, with our case utilizing the StyleGAN3-R variant. The training process's speed and efficiency are influenced by the *gpus* parameter, determining the number of Graphical Processing Units, and the *batch* parameter, which sets the global batch size for training samples processed in one cycle. *batch* controls the number of training samples the model trains to before updating the model's parameters. The model's generalization capability is impacted by the *gamma* hyperparameter, associated with learning rates and regularization. The right choice of this parameter depends on the size of our images, StyleGAN3 configuration, but for our case, we have followed this value to 2 as suggested in [4]. The $batch - gpu$ parameter ensures even distribution of the global batch size across GPUs, while *kimg* sets the total training duration, dictating the number of images the network encounters throughout training. StyleGAN3 logs the model performance in terms of ticks, and one tick is equivalent to 5 kimg in our experiment. We trained the StyleGAN3-R variant for 3900 kimg (or 780 ticks) with the snap of 100 kimg (or 20 ticks), although we manually checked the trained model every 6 hours to check the model's progress. The training took around three days on Sharcnet Graham computes [1].

- **Image Generation with StyleGAN3**: Generating images using StyleGAN3 is a nuanced process where specific parameters dictate the output's quality, variety, and attributes of the output. For instance, the *seeds* parameter acts like a series of unique keys or starting points, guiding the random number generator in creating diverse images. By tweaking these seed values, we can generate a wide array of images, each with its unique characteristics. The $truncation-psi$ parameter is akin to a balancing act between familiarity and novelty. At a lower setting, it ensures the generated images strongly resemble the training dataset, providing a sense of familiarity. However, as its value increases, the generated images exhibit more novelty, introducing diverse yet coherent variations. The $num-images$ parameter straightforwardly determines the volume of the image output, allowing for batch generation, which is invaluable when one requires a collection of images for assessments or choices. Lastly, the optional *quality* parameter serves as a trade-off knob between image quality and computational demands. Adjusting this can lead to higher fidelity images but might demand more computational power and time. The above parameters can be fine-tuned to generate images of sober faces with marijuana-intoxicated eyes. The images in the MID2023 are created by including cropped sections focusing on the eyes from generated images. Additionally, this dataset incorporates images from the previous dataset [20]. To augment the current MID2023 dataset further, conventional augmentation techniques, including flipping, cropping, and rotating images, have been employed.

5.3 Results and Discussion

StyleGAN-R was trained on 2.86M parameters for 3900 kimg with a snap of 100 kimg, while the training time was three days on two nodes of Sharcnet Graham P100 Pascal GPUs [1] with 50 GB RAM. During the training, we manually checked the model's progress by looking at the quality of images generated every 6 hours. Firstly, we experimented for two days, at which StyleGAN3-R had an FID of 35.20 at 2400 kimg (or 480 ticks). Figure 15 shows 4800 images generated by the StyleGAN3-R variant after two days. FID was close to 35 after the first training, which means the model is still learning. Later, we conducted the same experiment with the trained StyleGAN3-R variant for another 24 hours. After three days of training, we achieved

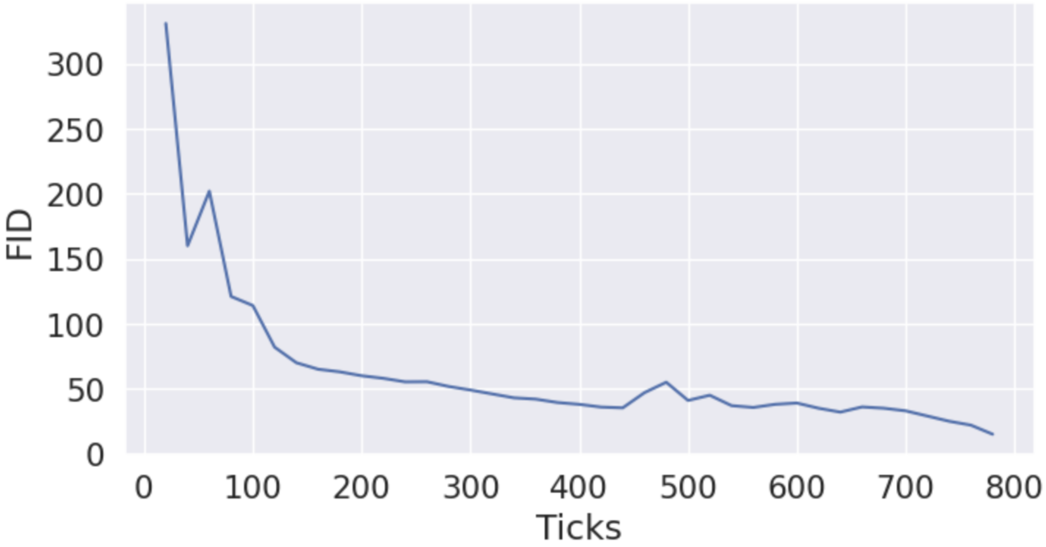


Fig. 13: Tick vs FID in StyleGAN3.

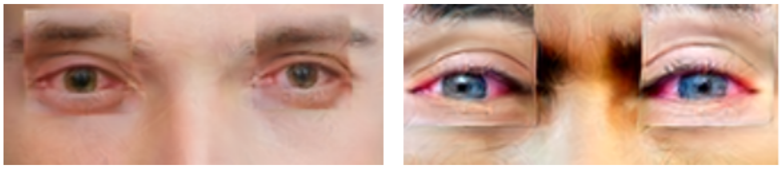


Fig. 14: Marijuana intoxicated eyes generated by StyleGAN3.

an FID score of 12.5. Figure 13 shows the FID at every tick or 5 kimg. StyleGAN3 logs the model performance in terms of ticks, and one tick is equivalent to 5 kimg in our experiment. Figure 13 shows that a peak occurred around 480 ticks because we retrained the model after two days, which reduced model performance.

After this experiment, StyleGAN3 generated an additional 600 images for each category, effectively doubling the original count and contributing a total of 1200 synthetic images. Moreover, classical augmentation techniques—which include rotation, flipping, scaling, cropping, and color adjustment—have been applied to increase the MID2023 further.

These techniques have produced a substantial number of augmented images: 1500 for each class, adding 3000 classical augmented images to the pool. Thus, the final tally of augmented images stands at 4200. When considering both the original and augmented images, the dataset now comprises 4800 images. MID2023 is divided into two classes: 2400 images of sober eyes and 2400 images of marijuana-intoxicated

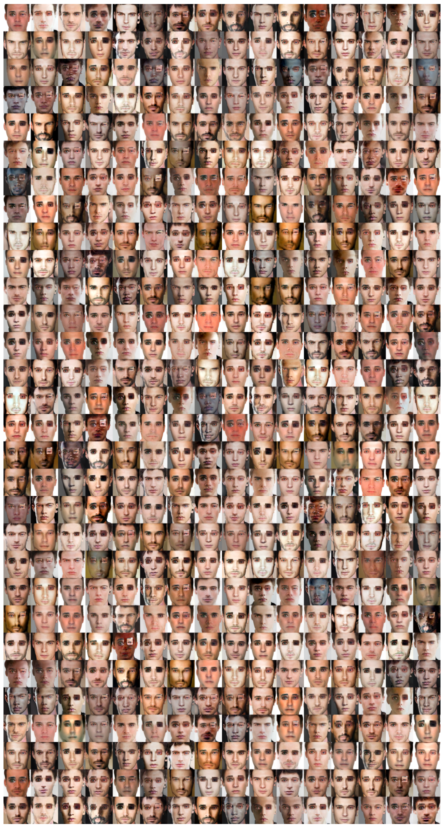


Fig. 15: Images generated by StyleGAN3-R at 2400 kimg.

eyes. Figure 14 shows two intoxicated red eyes generated from the StyleGAN3 model. Figure 14 also shows that StyleGAN3 can generate a similar yet distinctive set of images. Moreover, both images have different levels of intoxication, which can be seen with the intensity of redness. However, the influence of MID2023 will be validated in the later chapter by comparing the performance of the MobileNet, VGG-16, ResNet-50 and Inception-v3 model [20] with MID2021 and MID2023.

CHAPTER 6

Enhancement of Marijuana Intoxication Detection Models

In the last chapter, we augmented the MID2021 to generate MID2023. To validate the improvement in the MID2023, we will compare the performance of the MobileNet model trained on MID2023 with the baseline model, MobileNet [20] trained on MID2021. Moreover, an examination of the performance metrics, including accuracy, precision, recall, and F1-score, was carried out for three other models—VGG-16, ResNet-50, and Inception-v3—trained on both the augmented MID2023 dataset and the original MID2021 dataset. The evaluation was performed on separate test sets associated with MID2023 and MID2021, respectively. Section 6.1 discusses in detail our experiments, which include details of the dataset and the evaluation metrics utilized. Section 6.2 details the pipeline we implemented to evaluate VGG-16, ResNet-50, Inception-v3, and MobileNet on MID2023. Section 6.3 lists and discusses the results obtained from the experiments.

6.1 Dataset and Evaluation Metrics

A total of 480 experiments were conducted in tuning the VGG-16, ResNet-50, MobileNet and Inception-v3 marijuana-intoxication detection model with MID2023. This thesis compares the performance of the above models tuned on MID2023 with the baseline MobileNet model [20] tuned on MID2021. The subsequent sub-section discusses the MID2023 used in training marijuana-intoxicated models and section 6.1.2

elaborates on performance metrics including accuracy, precision, recall, and F1-score, employed for evaluating marijuana-intoxication detection models.

6.1.1 Dataset

MID2023: MID2023 consists of 4800 images, including 600 original eye images sourced from the original eye images in MID2021, 1200 images generated by StyleGAN3 and 3000 images generated by using traditional augmentation on both the original eye images and images generated by StyleGAN3. This dataset is equally split into two categories: 2400 images of bloodshot eyes, suggestive of marijuana intoxication, and 2400 images of sober eyes. Table 3 shows the number of original, augmented and total images in the MID2023. We can infer from the table 3 that two classes are balanced as stoned set and sober set have an equal number of images.

Number of Original Sober Images	300
Number of Original Intoxicated Images	300
Number of Original Images	**600**
Number of StyleGAN3 Augmented Sober Images	600
Number of StyleGAN3 Augmented Intoxicated Images	600
Number of Traditional Augmented Sober Images	1500
Number of Traditional Augmented Intoxicated Images	1500
Number of Augmented Images	**4200**
Number of Total Images	**4800**

Table 3: MID2023 Summary

6.1.2 Evaluation Metrics

Evaluation metrics are quantitative measures that are used to assess the performance of a machine learning model. To access the performance of CNN models for binary classification, we employed accuracy, precision, recall, and F1-score as evaluation metrics.

Accuracy: Accuracy is the ratio of the number of true predictions to the total number of predictions made by the system.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where,

TP (True Positive): Total number of images correctly predicted as intoxicated,

TN (True Negative): Total number of images correctly predicted as sober,

FP (False Positive): Total number of sober images incorrectly predicted as intoxicated, and,

FN (False Negative): Total number of intoxicated images incorrectly predicted as sober.

Precision: Precision is the ratio of true positive predictions to the total number of positive predictions.

$$Precision = \frac{TP}{TP + FP}$$

Recall: Recall is the ratio of the total number of positive predictions which are correct to the total number of predictions which are actually positive.

$$Recall = \frac{TP}{TP + FN}$$

F1 Score: In our prior discussions, we delineated the significance of both precision and recall as individual metrics. Nonetheless, to comprehensively evaluate a model's proficiency, it is imperative to account for both metrics concurrently. Within machine learning, the F1 Score emerges as a pivotal metric that integrates precision and recall. Defined as the harmonic mean of these two metrics, the F1 Score offers a nuanced understanding of model performance. Specifically, in the context of discerning marijuana-intoxicated imagery, the F1 Score quantifies the model's capability

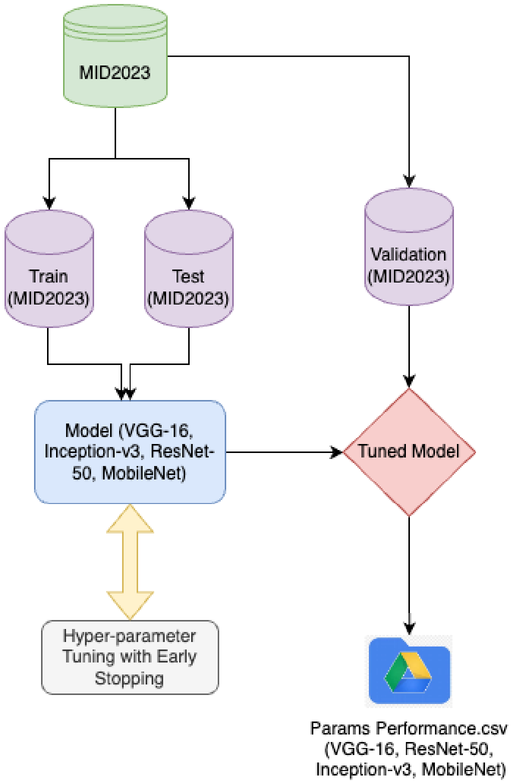

MID2023
Train (MID2023)
Test (MID2023)
Validation (MID2023)
Model (VGG-16, Inception-v3, ResNet-50, MobileNet)
Tuned Model
Hyper-parameter Tuning with Early Stopping
Params Performance.csv (VGG-16, ResNet-50, Inception-v3, MobileNet)

Fig. 16: Pipeline for fine-tuning marijuana intoxication detection models.

to accurately identify such images while mitigating the incidence of both false positives (erroneous categorization of marijuana-intoxicated images as sober) and false negatives (misclassification of sober images as marijuana-intoxicated).

$$F1\ Score = 2 \text{ x } \frac{Precession \text{ x } Recall}{Precession + Recall}$$

6.2 Pipeline for Fine-tuning CNN Models

Figure 16 shows the pipeline developed to store the performance of each model trained on MID2023 with 120 different hyperparameters. Firstly, we will discuss the initial setup for each of the models, and then we will discuss how we will tune each model with different hyper-parameters. Moreover, the results for each hyper-parameter are stored in Google Drive as a comma-separated value file.

6.2.1 Model Setup

CNNs have revolutionized image-based classification tasks. In this study, we leveraged pre-trained models of VGG-16, ResNet-50, and Inception-v3 to assess their effectiveness in detecting signs of marijuana use, particularly bloodshot eyes. Moreover, VGG-16, ResNet-50, and Inception-v3 had achieved accuracy of 95.48%, 95.78% and 97.08%, respectively for eyes disease detection [11]. VGG-16 [19] [52] is a widely used algorithm for both object detection and image classification, capable of categorizing images into 1,000 different classes with an accuracy of 92.7%. It is particularly user-friendly for transfer learning applications. Inception V3 [55], a convolutional neural network (CNN) architecture, was developed by researchers at Google and represents a significant milestone in the evolution of deep learning models for image classification. Inception V3 was introduced to address the growing demand for more efficient and accurate image recognition algorithms. ResNet-50 [22] is a revolutionary neural network model primarily designed for image classification, part of the ResNet family, which also addresses the vanishing gradient problem in deep neural networks. The model is widely recognized for its residual learning framework, which facilitates the training of extremely deep networks. Utilizing pre-trained models allows us to focus on adjusting the model's existing parameters to adapt to our specific task rather than training from scratch. Moreover, we build the MobileNet network as mentioned in [20] to develop the baseline for this study. The following sub-sections will explain the process of the initial model setup for the pre-trained CNN models and the MobileNet model.

- **VGG-16:** This model is renowned for its simplicity and depth, leveraging pre-trained weights derived from the extensive ImageNet database to harness its powerful feature extraction capabilities. The feature extraction layers are frozen to preserve the intricate high-level features already learned, ensuring stability and specificity in feature detection. For the task at hand, a custom fully connected network replaces the original classifier, specifically designed to discern between sober and intoxicated states. This network is further enhanced with

ReLU activation functions and dropout layers, introducing non-linearity and mitigating potential overfitting, thus ensuring a robust and generalized learning. The model employs the Stochastic Gradient Descent (SGD) optimizer, which is celebrated for its efficiency and effectiveness, particularly in large-scale and complex datasets. The Binary Cross Entropy Loss function is meticulously chosen to accurately measure the performance of the binary classification, providing a clear and quantifiable metric for model optimization.

- **ResNet-50:** As a deeper architecture with residual connections, ResNet-50 is adept at learning from a substantially increased number of layers without succumbing to the vanishing gradient problem. It starts with a foundation of pre-trained weights from ImageNet, which encapsulates a wide variety of features applicable to a wide range of visual recognition tasks. The model's layers are frozen to retain the nuanced feature detectors developed through extensive prior training. A bespoke classifier, specifically engineered for the binary classification task, is appended to the model. This classifier benefits from the introduction of ReLU activation functions and dropout layers, which collectively enhance the model's ability to learn complex patterns while avoiding overfitting. The AdamW optimizer is employed, known for its adaptive learning rate capabilities and effective weight decay, which are crucial for navigating the complex landscape of high-dimensional weight space. Furthermore, the StepLR learning rate scheduler is utilized, methodically decaying the learning rate at predetermined intervals to refine the convergence process and explore a more diverse set of potential solutions.

- **Inception-v3:** Inception-v3 stands out with its unique architecture that incorporates multiple kernel sizes within the same network layer to capture a diverse range of features. The model, pre-loaded with weights trained on the comprehensive ImageNet dataset, benefits from a broad and versatile feature extraction base. By freezing these layers, Inception-v3 maintains the rich feature understanding previously acquired, allowing it to focus on learning the specifics of the new task. The classifier is carefully restructured to cater to the binary classification of sober versus intoxicated states, with additional layers such as ReLU for introducing non-linearity and dropout for preventing overfitting, thereby ensuring a balanced and generalizable model. The adoption of the AdamW optimizer offers an advanced approach to adjusting weights through adaptive learning rates and weight decay, enhancing the training stability and outcome. The StepLR scheduler complements this by periodically adjusting the learning rate, aiding in the avoidance of local minima and promoting a thorough exploration of the weight space for optimal performance.

- **MobileNet:** The MobileNet network comprises two convolutional layers, each with 32 filters of size 3x3, a stride of 1 and padding of 1 to maintain the spatial dimensions of the input after convolution. Following the convolutional layers is a max-pooling layer with a 2x2 window and a stride of 2, which reduces the spatial dimensions of the feature maps by half. Max-pooling is a typical operation used to reduce the number of parameters and computations in the network and also to control overfitting by providing an abstracted form of representation. The network then flattens the output of the pooling layers and passes it to a fully connected layer. This is followed by a dropout layer with a probability of 0.5, which randomly zeros some of the elements of the input tensor with the given probability, thus helping to prevent overfitting by forcing the network to learn redundant representations. Since the problem is a binary classification, the Binary Cross Entropy Loss function is used as the criterion, suitable for outputs that are probabilities of two classes. The optimizer used is RMSprop, a type of adaptive learning rate method that divides the learning rate for a weight by a running average of the magnitudes of recent gradients for that weight. This can help to accelerate convergence in the right direction and is often used in recurrent neural networks and other contexts. The final layer is a fully connected layer that classifies if the image is sober or intoxicated.

6.2.2 Tuning Model

This section discusses the approach followed in tuning hyper-parameters for four models: ResNet-50, Inception-v3, VGG-16 and MobileNet. We divided MID2023 and MID2021 into three sets, training (70%), validation (20%), and test (10%). We iterated over a pre-defined grid of hyper-parameters that includes variations in batch sizes for training and validation, epochs, a gamma parameter influencing learning rate adaptation or regularization, and the initial learning rate. The comprehensive search includes a pre-planned set of combinations derived from lists of batch sizes of 8, 16, 24, 32 for training and 24, 16, 8 for validation, two gamma values of 0.1, 10, and five different learning rates of 0.01, 0.005, 0.0001, 0.00005, 0.000005, creating 120 permutations to validate. In each iteration, each model trains using the current set of hyper-parameters, evaluating it on the test set to obtain metrics like accuracy, recall, precision, and the F1-score. These metrics are crucial, with the F1-score being particularly important as it provides a harmonic balance between precision and recall, beneficial for class-imbalanced datasets. A dictionary meticulously documents each hyper-parameter trial's performance, creating a unique identifier for easy retrieval. This exhaustive process seeks the combination that yields the highest F1-score and integrates an early stopping mechanism. Early stopping monitors the validation loss, halting training if the model ceases to improve, effectively preventing over-fitting and saving computational resources. Upon completion, the function reports the optimal hyper-parameter set, using the F1-score as the selection criterion. This optimal set is expected to offer the most effective model configuration for the given data and task. Results from all combinations are stored in CSV files, ensuring that the data is accessible for further analysis or review. This systematic strategy encapsulates the quest for the best model performance and the optimal use of resources via early stopping, showcasing a balanced approach to machine learning model development.

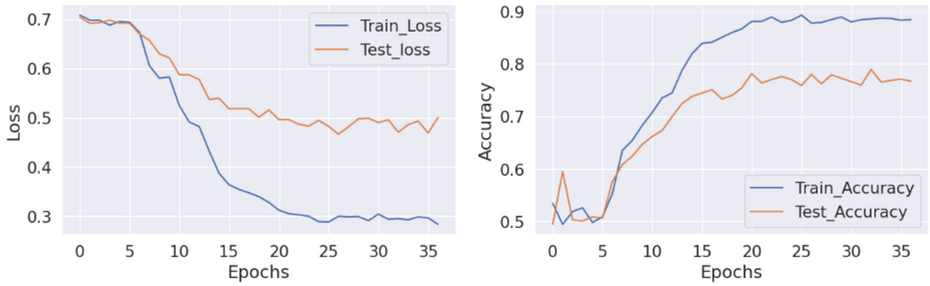


Fig. 17: Accuracy and Loss over Epochs for best MobileNet Model on MID2023.

6.3 Results and Discussion

For our experiments, we evaluated and compared the performance of three state-of-the-art models, VGG-16, ResNet-50 and Inception-v3, with the MobileNet model trained on MID2023 as well as MID2021.

We first try to compare the performance of our dataset by training the MobileNet model [20] on MID2023 and compare its performance with MobileNet trained on MID2021. This way, we can evaluate if augmenting images with StyleGAN3 helps improve the model's performance. We conducted almost 120 experiments on the MobileNet model using grid search on hyper-parameters, gamma, learning rate, training batch size and text batch size. Out of 120 hyper-parameter configurations, we achieved the best results for configuration in which training batch size was 16, validation batch size was 24, gamma was 0.1, and learning rate was 0.01. Figure 17 shows the loss and accuracy over epochs with the best hyper-parameters for the MobileNet model.

Now, we have validated that the performance of CNN models improved by using the StyleGAN3 image augmentation. We will validate the performance of the VGG-16, ResNet-50 and Inception-v3 with MID2023.

Comparing the performance of the MobileNet on the MID2021 and MID2023 datasets, there's a noticeable improvement over time. The MID2021 dataset yielded an accuracy of 82%, precision of 80%, recall of 83%, and an F1-score of 81%. In contrast, the MID2023 dataset shows enhanced performance, with accuracy increasing to 86.54%, precision to 82.74%, recall slightly decreasing to 79.12%, and an im-

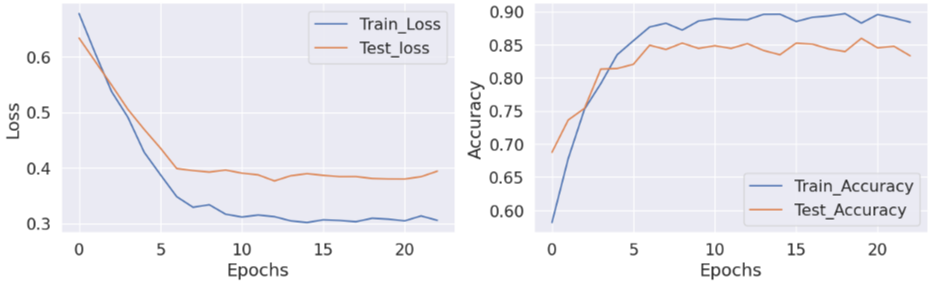


Fig. 18: Accuracy and Loss over Epochs for best ResNet-50 on MID2023.

proved F1-score of 84.89%. The increase in accuracy and F1 score indicates that the model has become better at correctly classifying images and maintaining a balanced precision-recall trade-off despite a slight drop in its ability to identify all actual positives (recall).

Model	**Training Batch Size**	**Validation Batch Size**	**Gamma**	**Learning Rate**
VGG-16	8	8	0.1	5e-05
ResNet-50	24	24	0.1	5e-05
Inception-v3	8	24	0.1	1e-04
MobileNet	16	24	0.1	1e-02

Table 4: Best Hyper-parameters for models trained on MID2023

Table 4 presents the optimal hyper-parameter configurations for various convolutional neural network models derived from systematic tuning to achieve the best performance on a MID2023. For VGG-16, the ideal training and validation batch sizes were both set to 8, with a gamma value of 0.1 and a learning rate of 5×10^{-5}, suggesting a preference for smaller batch processing and moderate regularization during training. The ResNet-50 model operates best with uniform batch sizes 24 for both training and validation sets, coupled with a gamma of 0.1 and the same learning rate as VGG-16, indicating its effective learning from larger batch sizes without altering the learning rate. In contrast, Inception-v3 shows an optimal performance with a training batch size of 8 and a larger validation batch size of 24, alongside a higher learning rate of 1×10^{-4}, which could imply a need for more rapid adjustments during learning, balanced by a stable evaluation phase with larger test batches. The

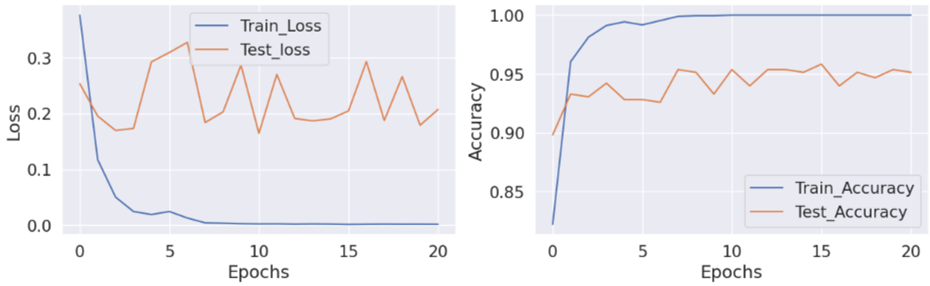


Fig. 19: Accuracy and Loss over Epochs for best VGG-16 on MID2023.

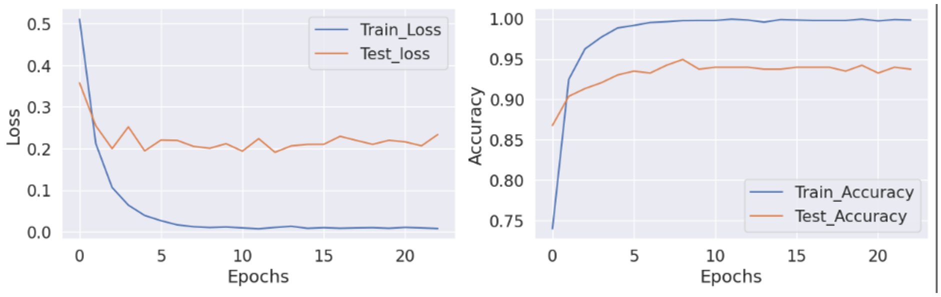


Fig. 20: Accuracy and Loss over Epochs for best Inception-v3 on MID2023.

MobileNet model, with its mid-range training and validation batch sizes of 16 and 24, respectively, and a relatively high learning rate of 1×10^{-2}, reflects a distinct configuration that possibly accommodates its architectural differences and learning dynamics. These tailored hyperparameters underscore the importance of fine-tuning learning processes to the specific characteristics of each neural network architecture to enhance model performance. Figure 19, 20, and 18 shows the loss and accuracy over epochs with best hyper-parameters for VGG-16, Inception-v3, and ResNet-50 respectively.

Table 5 provides a comparative summary of performance metrics for different marijuana intoxication models evaluated on the MID2023 and MID2021 datasets. For MID2023, the VGG-16 model demonstrates superior performance with the highest accuracy (94.66%), precision (96.84%), and F1-score (92.92%), indicating its robustness in correctly identifying and classifying images with precision. Moreover, Inception-v3 also shows commendable results, particularly with the highest recall (95.12%), suggesting its strength in identifying relevant cases within MID2023 dataset. However, ResNet-50, while presenting moderate accuracy (82.34%), has lower precision and

Model	**Dataset**	**Accuracy**	**Precision**	**Recall**	**F1-score**
VGG-16	MID2023	**94.66**	**96.84**	89.32	**92.92**
VGG-16	MID2021	75.23	76.45	78.92	76.43
ResNet-50	MID2023	82.34	71.49	79.12	75.11
ResNet-50	MID2021	71.26	73.85	76.57	69.11
Inception-v3	MID2023	90.18	82.1	**95.12**	88.13
Inception-v3	MID2021	80.21	80.51	82.63	81.55
MobileNet	MID2023	86.54	82.74	79.12	84.89
MobileNet	MID2021	82	80	83	81

Table 5: Best Performance for Marijuana Intoxication Models

recall, leading to the lowest F1-score (75.11%) among the models tested on MID2023, which could indicate a relatively lesser ability to balance the trade-off between precision and recall. Notably, the performance of VGG-16, ResNet-50, and Inception-v3 exhibited suboptimal results when trained on the MID2021 dataset as opposed to their performance on the augmented MID2023 dataset. This discrepancy can be attributed to the inherent demand for a substantial volume of data by these larger neural network architectures, a requirement more adequately fulfilled by the enriched MID2023 dataset. Lastly, the MobileNet model's performance is consistent across both datasets, with improvements in accuracy, precision and F1-score when transitioning from MID2021 to MID2023, reflecting advancements in model training or dataset quality. The increased recall observed in MobileNet for MID2021 could be attributed to its compact network architecture. However, this recall diminishes in MID2023, which indicates the model's limited capacity to handle the diverse instances present in the MID2023 effectively. The next chapter will present conclusions drawn from the study's findings, accompanied by a discussion on potential avenues for future research.

CHAPTER 7

Conclusion and Future Work

7.1 Conclusion

This thesis has created a new dataset MID2023 by using StyleGAN3. Based on our experiment of MobileNet with MID2021 and MID2023, MID2021 has a slightly higher recall, while, MID2023 outperforms MID2021 in terms of accuracy, precision, and F1-score. This clearly shows that MID2023 is inhancement of MID2021.

Moreover, we optimized the performance of VGG-16, ResNet-50 and Inception-v3 models for marijuana intoxication detection by tuning these models with 120 different hyper-parameters. Among the models assessed, the VGG-16 model emerged as the top-performing candidate, surpassing the reference MobileNet model across multiple key metrics. In contrast, other models such as ResNet-50 and Inception-v3 did not perform as well as VGG-16. ResNet-50 displayed lower accuracy, precision, and F1-score, while, Inception-v3 had slightly lower accuracy and F1-score compared to VGG-16. MobileNet also fell short in terms of overall performance compared to VGG-16.

In conclusion, the VGG-16 model's superior performance, as evidenced by its exceptional accuracy, precision, recall, and F1-score, establishes it as the most effective choice among the evaluated models in comparison to the MobileNet model for the given dataset and task.

7.2 Future Works

Given that red eyes are a crucial indicator for identifying marijuana intoxication, the model must account for various eye-related conditions that might also cause redness, thereby preventing misinterpretation and reducing false positives. For instance, conditions like conjunctivitis or dry eye syndrome can lead to similar red-eye symptoms. Therefore, incorporating a mechanism to differentiate between the redness caused by eye-related conditions and that induced by marijuana intoxication is crucial for the accuracy and reliability of the model's assessments.

In their examination of driver drowsiness detection methodologies, Albadawi et al. [5] noted that the onset of driver drowsiness is typically preceded by observable symptoms. These symptoms range from difficulty maintaining eye openness, increased yawning, and frequent blinking to challenges in concentration, lane deviations, delayed traffic responses, head nodding, and erratic speed changes. To enhance the practical application of a marijuana intoxication model, it is essential to conduct empirical research involving actual human subjects. This research would involve capturing images of participants in a sober state and subsequently after marijuana consumption. Such a methodology ensures that a diverse range of intoxication indicators are recorded, potentially enhancing the model's accuracy and performance.

REFERENCES

[1] (2017). Graham - alliance doc.

[2] (2019). Cannabis (marijuana) drugfacts. *Journal of Electronic Imaging*.

[3] Aberman, K., Weng, Y., Lischinski, D., Cohen-Or, D., and Chen, B. (2020). Unpaired motion style transfer from video to animation. *ACM Transactions on Graphics (TOG)*, 39(4):64–1.

[4] Alaluf, Y., Patashnik, O., Wu, Z., Zamir, A., Shechtman, E., Lischinski, D., and Cohen-Or, D. (2022). Third time's the charm? image and video editing with stylegan3. In *European Conference on Computer Vision*, pages 204–220. Springer.

[5] Albadawi, Y., Takruri, M., and Awad, M. (2022). A review of recent developments in driver drowsiness detection systems. *Sensors*, 22(5).

[6] Albawi, S., Mohammed, T. A., and Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)*, pages 1–6. Ieee.

[7] Alsayadi, H., Abdelhamid, A., Hegazy, I., and Taha, Z. (2021). Data augmentation for arabic speech recognition based on end-to-end deep learning. *International Journal of Intelligent Computing and Information Sciences*, 21(2):50–64.

[8] Azcarate, P. M., Zhang, A. J., Keyhani, S., Steigerwald, S., Ishida, J. H., and Cohen, B. E. (2020). Medical reasons for marijuana use, forms of use, and patient perception of physician attitudes among the us population. *Journal of general internal medicine*, 35:1979–1986.

[9] Benalcazar, D. P., Benalcazar, D. A., and Valenzuela, A. (2022). Artificial pupil dilation for data augmentation in iris semantic segmentation. In *2022 IEEE Sixth Ecuador Technical Chapters Meeting (ETCM)*, pages 1–6. IEEE.

[10] Bengio, Y., Goodfellow, I., and Courville, A. (2017). *Deep learning*, volume 1. MIT press Cambridge, MA, USA.

[11] Bitto, A. K. and Mahmud, I. (2022). Multi categorical of common eye disease detect using convolutional neural network: a transfer learning approach. *Bulletin of Electrical Engineering and Informatics*, 11(4):2378–2387.

[12] Burns, S. (2019). *Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow.*

[13] Canada, H. (2023). About cannabis.

[14] Canadian Substance Use Costs and Harms Scientific Working Group (2023). Canadian substance use costs and harms 2007–2020. Technical report, Canadian Institute for Substance Use Research and the Canadian Centre on Substance Use and Addiction, Ottawa, Ont.

[15] Chlap, P., Min, H., Vandenberg, N., Dowling, J., Holloway, L., and Haworth, A. (2021). A review of medical image data augmentation techniques for deep learning applications. *Journal of Medical Imaging and Radiation Oncology*, 65(5):545–563.

[16] Darthmanav (2020). Resnet 50 , vgg 16, inception v3 -beginner's guide.

[17] DeGregorio, M. W., Wurz, G. T., Montoya, E., and Kao, C.-J. (2021). A comprehensive breath test that confirms recent use of inhaled cannabis within the impairment window. *Scientific reports*, 11(1):22776.

[18] Endo, Y. (2022). User-controllable latent transformer for stylegan image layout editing. In *Computer Graphics Forum*, volume 41, pages 395–406. Wiley Online Library.

[19] G, R. (2021). Everything you need to know about vgg16. `https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918`.

[20] Gadhiya, R. (2021). Marijuana intoxication detection using convolutional neural network.

[21] Hartman, R. L., Brown, T. L., Milavetz, G., Spurgin, A., Gorelick, D. A., Gaffney, G., and Huestis, M. A. (2015). Controlled cannabis vaporizer administration: blood and plasma cannabinoids with and without alcohol. *Clinical chemistry*, 61(6):850–869.

[22] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[23] Horev, R. (2019). Explained: A style-based generator architecture for gans - generating and tuning realistic...

[24] Jackson NJ, Isen JD, K. R. (2016). Impact of adolescent marijuana use on intelligence: Results from two longitudinal twin studies. *PNAS*.

[25] Joesph, A. (2018). Fda approves country's first medicine made from marijuana. `https://www.statnews.com/2018/06/25/fda-approves-countrys-first-medicine-made-from-marijuana/`.

[26] Johnson, O. E., Miskelly, G. M., and Rindelaub, J. D. (2022). Testing for cannabis intoxication: Current issues and latest advancements. *Wiley Interdisciplinary Reviews: Forensic Science*, 4(4):e1450.

[27] Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.

[28] Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks.

[29] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119.

[30] Khalid, I. A. (2021). Face landmark detection using python.

[31] Lee, S. (2021). Any tips on choosing the value for r1-regularization?

[32] Liu, Z., Luo, P., Wang, X., and Tang, X. (2018). Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11.

[33] Lu, Y. (2023). Style-based image manipulation using the stylegan2-ada architecture. *ACE*, 2:29–37.

[34] M., D. (2022a). The finger-to-nose test in dui investigations - shouse law group.

[35] M., D. (2022b). The horizontal gaze nystagmus test: Faulty science in dui investigations american courts.

[36] M., D. (2022c). The rhomberg balance test.

[37] Marouf, A. A., Mottalib, M. M., Alhajj, R., Rokne, J., and Jafarullah, O. (2022). An efficient approach to predict eye diseases from symptoms using machine learning and ranker-based feature selection methods. *Bioengineering*, 10(1):25.

[38] Mcvean, A. (2018). It's not the smoke from a joint that makes your eyes red. `https://www.mcgill.ca/oss/article/did-you-know/its-not-smoke-joint-makes-your-eyes-red`.

[39] Meier MH, Caspi A, A. A. (2023). Persistent cannabis users show neuropsychological decline from childhood to midlife. *National Library of Science*.

[40] Mishra, M. (2021). Convolutional neural networks, explained.

[41] of Justice Canada, D. (2018). Impaired driving laws. `https://www.justice.gc.ca/eng/cj-jp/sidl-rlcfa/index.html#a1`.

[42] Pflanzer, L. R. (2016). A drug derived from marijuana is working for people with a rare form of epilepsy — and the company's stock is taking off. `https://www.businessinsider.com/drug-derived-from-marijuana-to-treat-epilepsy-gw-pharmaceuticals-epidiolex-2016-6`.

[43] PI, M. G. and Wilson, J. (2020). Differences in cannabis impairment and its measurement due to route of administration.

[Poornima and Sakkari] Poornima, G. and Sakkari, D. S. Diagnosis of skin diseases based on deep learning and machine learning approach: Technical review.

[45] Prabhu, R. (2019). Understanding of convolutional neural network (cnn) - deep learning.

[46] Prince, S. J. D. (2023a). *Title of the Specific Chapter or Section*, chapter 3, page 29. MIT Press, Cambridge, MA. Figure 3.4: Depicting neural networks.

[47] Prince, S. J. D. (2023b). *Title of the Specific Chapter or Section*, chapter 3, page 26. MIT Press, Cambridge, MA. Figure 3.2: Family of function defined by equation 3.1.

[48] Prince, S. J. D. (2023c). *Title of the Specific Chapter or Section*, chapter 10, page 162. MIT Press, Cambridge, MA. Figure 10.2: 1D convolution with kernel size three.

[49] Prince, S. J. D. (2023d). *Title of the Specific Chapter or Section*, chapter 10, page 162. MIT Press, Cambridge, MA. Figure 10.10: 2D convolution applied to an image.

[50] Prince, S. J. D. (2023e). *Title of the Specific Chapter or Section*, chapter 15, page 276. MIT Press, Cambridge, MA. Figure 15.1: GAN mechanism.

[51] Robertson, M., Li, A., Yuan, Y., Jiang, A., Gjerde, H., Staples, J., and Brubacher, J. (2022). Correlation between oral fluid and blood thc concentration: a systematic review and discussion of policy implications. *Accident Analysis & Prevention*, 173:106694.

[52] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252.

[53] Spindle, T. R., Martin, E. L., Grabenauer, M., Woodward, T., Milburn, M. A., and Vandrey, R. (2021). Assessment of cognitive and psychomotor impairment, subjective effects, and blood thc concentrations following acute administration of oral and vaporized cannabis. *Journal of psychopharmacology*, 35(7):786–803.

[54] Staff, M. C. (2023). Drugs suppliment of marijuana. `https://www.mayoclinic.org/drugs-supplements-marijuana/art-20364974`.

[55] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

[56] T, A. N. (2021). Inception v3 model architecture.

[57] Xie, J., Ouyang, H., Piao, J., Lei, C., and Chen, Q. (2023). High-fidelity 3d gan inversion by pseudo-multi-view optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 321–331.

[58] Zhou, L. and Zhang, T. (2023). AttCST: attention improves style transfer via contrastive learning. *Journal of Electronic Imaging*, 32(3):033018.

VITA AUCTORIS

NAME:	Puneet Jain.
PLACE OF BIRTH:	Ludhiana, Punjab, India.
YEAR OF BIRTH:	1996.
EDUCATION:	University of Windsor, M.Sc. in Computer Science, Windsor, Ontario, 2024. Guru Nanak Dev Engineering College, B. Tech. Computer Science, Ludhiana, Punjab, India, 2017.