FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Assignment of master's thesis

Title:	Facial expression analysis from NIR image
Student:	Bc. Tomáš Kalabis
Supervisor:	Ing. Jan Hejda, Ph.D.
Study program:	Informatics
Branch / specialization:	Knowledge Engineering
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2023/2024

Instructions

The aim of this work is to design methods for determining facial expressions from images acquired by a camera in the near infrared (NIR) spectrum.

Implement the following steps:
1) Familiarize with the problem of face image acquisition in the visible/NIR spectrum.
2) Study methods for transforming images in the visible spectrum to NIR and vice versa using machine learning.
3) Do a research on methods for face detection in images and their subsequent classification based on facial expressions.
4) Propose at least two methods for face detection and facial expression classification from NIR image. Use available datasets and transformations for training.

Statistically evaluate and compare the accuracy and computational complexity of the proposed models.

Proposed references:
Wang, H., Zhang, H., Yu, L., Wang, L., & Yang, X. (2020, May). Facial feature embedded CycleGAN for VIS-NIR translation. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1903-1907). IEEE.
Ko, B. C. (2018). A brief review of facial emotion recognition based on visual information. sensors, 18(2), 401.
Mollahosseini, A., Hasani, B., & Mahoor, M. H. (2017). Affectnet: A database for facial

---

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 16 February 2023 in Prague.

expression, valence, and arousal computing in the wild. IEEE Transactions on Affective Computing, 10(1), 18-31.

Master's thesis

FACIAL EXPRESSION ANALYSIS FROM NIR IMAGE

Bc. Tomáš Kalabis

Faculty of Information Technology
Department of Applied Mathematics
Supervisor: Ing. Jan Hejda, Ph.D.
May 11, 2024

Czech Technical University in Prague
Faculty of Information Technology

Link to this thesis: Bc. Tomáš Kalabis. *Facial expression analysis from NIR image.* Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2021.

Contents

List of Figures

List of Tables

List of Code Snippets

I would like to express my sincere gratitude to my thesis supervisor, Ing. Jan Hejda, Ph.D., for his valuable cooperation, friendly approach, and guidance throughout the entire process. I am also immensely grateful to everyone who contributed to the creation of CustomDB dataset in any way possible. Additionally, I would like to extend my heartfelt thanks to my family, friends, and loved ones for their unwavering support and encouragement during this journey.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as a school work under the provisions of Article 60 (1) of the Act.

In Prague on May 13 2020 .....

Abstract

Facial expression analysis (FER) has long been a significant subject in the field of computer vision, with a focus on predicting discrete emotional states and labels within the Circumplex model of affect - a sophisticated approach to capturing mental states. However, the effectiveness of images in the visible spectrum diminishes under poor lighting conditions. This is where near-infrared (NIR) images, which are resilient to changes in illumination, become crucial. Given the lack of an adequate dataset for FER from NIR images, the AffectNet dataset was converted to the NIR spectrum using the CycleGAN model, capable of transitioning from NIR to visible spectrum and vice versa. Additionally, modest custom datasets were also developed, which with other NIR datasets for FER were merged into a combined one. Several strategies were proposed to address the problem, encompassing face detection, image spectrum translation, and facial expression analysis using architectures such as CenterFace, RetinaFace, CycleGAN, MobileNet and DDAMFN. The entire system was encapsulated into a user-friendly Python module. The results demonstrated that the proposed solutions marginally surpassed the original solution designed for VIS images, which this work is based on. Notably, the facial expression recognition on the artificially transformed AffectNet dataset achieves near state-of-the-art (SOTA) performance. Furthermore, the facial expression recognition from NIR images surpasses the results of comparable studies. However, the performance on the combined true dataset is slightly lower in the discrete emotion predictions, which could be attributed to this dataset's small size and inherent challenges. There is a need for further testing on high-quality annotated NIR images with facial expressions. Despite these challenges, this work presents promising advancements in the field of facial expression analysis from NIR images.

Keywords Machine Learning, Facial Expression Analysis (FER), Image Spectrum Translation, Image Colourization, Face Detection, Near-Infrared (NIR) Images, Circumplex Model of Affect, CycleGAN, MobileNet, Dynamic Dual-Attention Multi-Face Network (DDAMFN), CenterFace, RetinaFace, OuluCasia, AffectNet, Image Processing, Artificial Intelligence

Abstrakt

Analýza výrazu obličeje (FER) je již dlouho významným tématem v oblasti počítačového vidění se zaměřením na předpovídání diskrétních emočních stavů a predikci hodnot v Circumplex Model of Affect modelu - sofistikovaného přístupu k zachycení duševnho rozpoložení. Použitelnost snímků ve viditelném spektru se však za špatných světelných podmínek snižuje. Pro řešení tohoto problémů jsou vhodné snímky z blízkého infračerveného (NIR) spektra, které jsou odolné vůči změnám osvětlení. Vzhledem k tomu, že neexistuje dostatečný dataset pro FER z NIR snímků, byl dataset AffectNet převeden do NIR spektra pomocí modelu CycleGAN, který je schopen přeložit snímky z NIR do viditelného spektra a naopak. Kromě toho byly vyvinuty také vlastní datasety, které byly s ostatními NIR datasety pro FER sloučeny do jendoho kombinovaného. K řešení problému bylo navrženo několik strategií zahrnujících detekci obličeje, převod spektra obrazu a analýzu výrazu obličeje s využitím architektur CenterFace, RetinaFace, CycleGAN, MobileNet a DDAMFN. Celý systém byl zapouzdřen do uživatelsky přívětivého Python modulu. Výsledky ukázaly, že navržené řešení mírně překonalo původní řešení určené pro snímky

ve viditelném spektru, z něhož tato práce vychází. Navíc, rozpoznávání výrazu obličeje na uměle transformovaném datasetu AffectNet dosahuje téměř nejlepších výsledků (SOTA). Rozpoznávání výrazu obličeje z NIR snímků navíc překonává výsledky srovnatelné studie. Výkonnost na kombinované sadě skutečných dat je však o něco nižší v oblasti predikce diskrétních emocí, což lze přičíst malému rozsahu této sady dat a vnitřním problémům. Je třeba provést další testování na kvalitněji anotovaných NIR snímcích s výrazy obličeje.

Klíčová slova Strojové učení, Analýza výrazu tváře (FER), Překlad spektra obrazu, Kolorizace obrazu, Detekce obličeje, Snímky v blízkém infračerveném spektru (NIR), Circumplex Model of Affect, Cycle Generative Adversarial Networks (CycleGAN), MobileNet, Dynamic Dual-Attention Multi-Face Network (DDAMFN), CenterFace, RetinaFace, OuluCasia, AffectNet, Zpracování obrazu, Umělá inteligence

List of Acronyms

Adam	Adaptive Moment Estimation
AI	Artificial Intelligence
ANN	Artificial Neural Network
CNN	Convolutional Neural Networks
CycleGAN	Cycle Generative Adversarial Network
DDAMFN	Dual-Direction Attention Mixed Feature Network
DL	Deep Learning
DNN	Deep Neural Network
FER	facial expression recognition
FDR	False Discovery Rate
FFE	facial feature extractor
FNN	Feedforward Neural Network
FNR	False Negative Rate
GAN	Generative Adversarial Network
MAE	Mean Absolute Error
ME	micro expression
ML	Machine Learning
MLP	Multilayer Perceptron
MSE	Mean Squared Error
NAS	Neural Architecture Search
NIR	Near-infrared
PPV	Positive Predictive Value
RGB	red-green-blue
RMSE	Root Mean Squared Error
SGD	Stochastic Gradient Descent
SOTA	State of the art
TPR	True Positive Rate
VA	Valence Arousal
VIS	Visible light

Introduction

The study of facial expressions has long captivated the fields of psychology, computer science and human-computer interaction. The ability to decipher the subtle nuances of human emotion, as conveyed through facial expressions, is a profound aspect of human cognition and communication. Understanding and decoding these expressions from an image with a computer system can provide a form of surveillance in a wide variety of fields including healthcare and daily life, or facilitate work such as in psychology research, which happens to be the application field of this thesis.

The system implemented in this thesis aims to be used in psychology research, specifically for monitoring the mental state of subjects in over-pressure chambers. In this study, they will be used for accommodating a water-like pressure in a so-called caisson, a watertight-box-like structure commonly used in water constructions. Over-pressure chambers are vessels where people can stay to prevent decompression sickness, which could arise when a sudden change of surrounding pressure such as in construction works in a caisson. People spend in the confined chamber several hours gradually accommodating to target pressure, which creates a unique psychological state since the ICE (isolated, confined and extreme) environment. The monitoring itself will be used for determining the impact of a subject's stay in the chamber on their mental state.

Although the existing system works in the visible spectrum, it is essential to monitor persons' well-being with the near-infrared (NIR) spectral camera, since there is either sometimes insufficient illumination or not at all, such as at night. The NIR spectral emits special illumination, that is not visible to the human eye, and captures the reflections of the illumination back. That creates a more robust solution independent of visible light illumination.

The main goal of this work is to design and implement a system for determining the facial expressions of a subject from videos/images acquired by a camera in the NIR spectrum. This consists of 2 separate working systems, one for *face detection* in the NIR spectrum and the second for *facial expression recognition*[1] from a NIR image. Each separate system will employ deep learning algorithms and will utilize provided data.

Additionally, a translation from the NIR spectrum to the visible spectrum will be explored and eventually designed, since there are not any sufficient datasets for facial expression analysis from the NIR spectrum.

Furthermore, the system will utilize an existing solution to create a more robust solution.

The thesis is structured in the following way. In the beginning, the first chapter discusses the research on the given topic. Follows the Computer Vision chapter introducing the field and delving into the theory of the used architectures. The third chapter, Methods, proposes the solution to the problem, designs the experiments and benchmarks and introduces novel modest CustomDB and CustomMorphSet datasets used in this thesis. Last but not least, it describes the Python module for inference of the whole system. The next chapter called Results reveals the outcomes of conducted experiments and benchmarks. The fifth chapter Discussion then debates

[1] Mentioned also as *facial expression analysis* (as in the title of the thesis) or *facial emotion recognition*.

and interprets the results and discusses unfinished and future work and the last chapter then concludes the whole thesis. Also, the appendix is enclosed for additional results.

Chapter 1
Analysis

This chapter researches the given topic and introduces the already existing solutions. That includes a description of available databases and an introduction of subtasks.

1.1 Acquisition process of facial expression from NIR images

To introduce the problem addressed in this thesis, the entire acquisition process must first be elaborated upon. Determining facial expressions from images acquired by a camera in the near-infrared (NIR) spectrum can be a complicated problem to tackle. First due to the fact, that a whole system needs to be compounded from multiple subsystems - the *facial expression recognition* system itself and the *face detection* system. Second is the fact, that there is not a sufficient dataset for training facial emotion recognition systems from NIR images [4]. That creates a need for another subsystem – *image spectrum translation* system. That can be either used for translation from the near-infrared spectrum to the visible spectrum (VIS) and subsequently classify facial emotion, or utilized for data preparation of facial emotion recognition system such as in [4].

Therefore, the whole system will comprise the following sub-systems:

- *Face detection system*,
- *Face image spectrum translation system*,
- *Facial expression recognition system itself.*

Those will be discussed in the next sections. The following subsections introduce the NIR spectrum and related work to this system.

1.1.1 NIR spectrum

The near-infrared (NIR) spectrum has emerged as a pivotal asset in advancing the field of medical diagnosis, industries such as the dairy industry [47] or face recognition and similar sub-fields such as facial emotion recognition (FER). That is due to the fact, that the VIS camera is insufficient while low or no illumination [31].

The wavelength of NIR images is between $0.7\mu m$ and $1.4\mu m$, and VIS images are between $0.4\mu m$ and $0.75\mu m$ [47][1]. This wavelength offers a distinct advantage in independence on il-

[1] NIR $0.7\mu m - 1.4\mu m$ wavelength is often referred to as short-wave NIR (SWIR). Another definition has a wavelength between $0.7\mu m - 2.5\mu m$ and is also called broadband NIR.

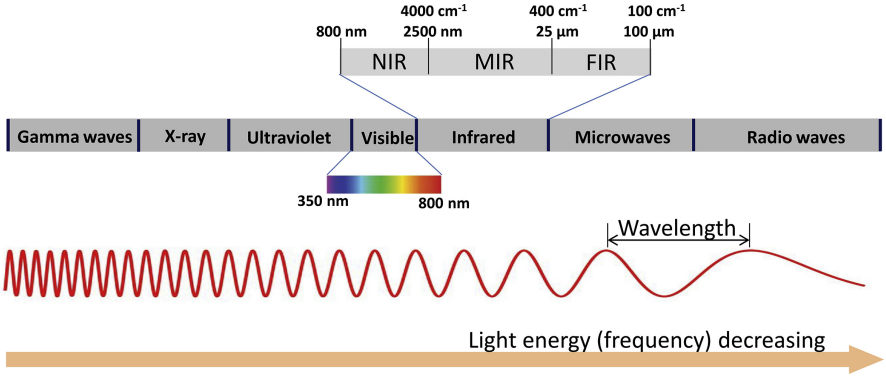


Figure 1.1 Depiction of broadband NIR1 and VIS spectrum on electromagnetic spectrum [47].

lumination, as mentioned earlier. Other infrared (IR) parts such as middle infrared provide functionality as well, NIR is a low-cost solution compared to other parts of an IR spectrum [22]. A figure 1.1 depicts the position of NIR and VIS on the electromagnetic spectrum.

1.1.2 Related work

In the paper [4], the researchers have proposed the following framework on how to classify facial expressions from NIR images. For assessing the facial expressions, they utilized a polar version of Russel's circumplex model of affect [54] rather than the traditional discrete concept of emotions, which are anger, disgust, fear, happiness, sadness, and surprise [2][11]. As authors say, traditional concept is "limited due to the coarse granularity of emotion labels and do not work well in real-world scenarios [69]". Figure 1.2 shows a polar version of a circumplex model, which maps emotions to a two-dimensional Valence-Arousal space, where valence indicates how positive or negative emotion is, and arousal distinguishes active from passive emotions.

At first, they trained the CycleGAN [81] neural network to create a model transforming from VIS spectrum images to NIR spectrum images. That was due to the fact, that there is not a sufficient dataset for training facial emotion recognition system from NIR image. This model was trained on existing morphed [3] Oulu-CASIA NIR&VIS facial expression database [79] which consists of paired images (one in visible spectrum and one in NIR spectrum).

With the help of this model, they have transformed an extensive morphed *AffectNet* database [42] and *MorphSet* database [68] of annotated facial expression images from the visible spectrum into the NIR spectrum. That was further used for training a facial expression classifier model – this classifier had the architecture of *EfficientNet-B0* [66].

In their conducted experiments, utilizing the MorphSet database outperformed usage of the *AffectNet* database [42]. Authors infer that it is due to the fact, that "both the Oulu-Casia dataset and VL MorphSet contain frontal face images taken in a lab setting, whereas AffectNet images are completely unconstrained".

Other studies focused on the given topic but they did not provide robust solutions employing only the OuluCasia dataset or were outdated [57][65][78].

[2] Or additionally contempt and neutral.

[3] They transformed the categorical dataset to a dimensional dataset with valence-arousal labels following the framework proposed in [68]. Also, this enabled them to augment the original Oulu-Casia Database consisting of 1 sample per emotion per subject to 10 samples. It was achieved by Delaunay triangulation followed by local warping of 68 facial landmarks from Dlib [29] face recognition system.

1.2 Face detection from NIR image

Face detection is a task in computer vision determining the existence of and corresponding location of human faces in digital images. Generally, there are 2 common approaches in face detection – feature-based and image-based [61]. The first approach is based on facial feature knowledge and it is typically very fast, however, the performance is typically worse [45] than the image-based approach. A typical example is the Viola-Jones algorithm5 [45] classifier available in the OpenCV library. The second approach is image-based, which employs neural networks, has typically better performance compared to the feature-based approach. One of the existing models for face detection is YOLO-face, whose architecture is based on YOLOv3 [61]. It has the same detection speed but it solves the problem of varying faces [48][5]. Another model worth mentioning is RetinaFace [7], part of the InsightFace project. It is a single-stage face detector that performs pixel-wise face localization on various scales of faces. It is known for its impressive detection performance even in crowded scenes. Moreover, RetinaFace can run real-time on a single CPU core for a VGA-resolution image, making it suitable for mobile devices. Another technique is MultiTask Cascaded CNN (MTCNN) which is a robust and fast detector [61][76], which offers also facial landmark detection and facial alignment. The face detection is achieved by implementing a cascade of 3 CNNs. Nevertheless, the face detector used in an existing solution is a CenterFace [73] from Xu *et al.* (2020), which can achieve superior accuracy and speed, hence is suitable for mobile devices and can run on CPU cores.

1.3 Image spectrum translation

This section describes an image spectrum translation task and focuses specifically on translation from the NIR spectrum to the VIS spectrum and vice versa. Also, a few notable studies are described.

The image spectrum translation is a type of task often referred to as “image-to-image translation” or “domain adaptation”, which is a domain of several types of GAN-based architectures [16] such as CycleGAN [81], Pix2Pix [25] or StyleGAN2 [28]. Additional types of neural network architectures that can be utilized for image-to-image translation include Variational Autoencoders (VAE) [38], Deterministic Variational Autoencoders (DVAE) [56], Vision Transformers (ViT) [8], and TransGAN [27]. In the case of NIR to VIS translation, this task is also referred to as “image colorization” [2]. Thus, the main goal is to translate the 1-channel grayscale image to the 3-channel RGB image. And conversely, the translation from VIS to NIR is a transfer from 3-channel RGB to 1-channel grayscale. Several studies focused directly on this use case or very similar use cases, thus translating from NIR to VIS (or vice versa) of face images.

Xu *et al.* (2021) introduced their method named *DenseUnet GAN*. Novel architecture synthesized well-established UNet [51] and DenseNet [23] architectures to use them as the Generator module. Additionally, they introduced more suitable loss functions for face image translation. This new architecture was compared with several suitable architectures - CycleGAN, Pix2Pix, Asymmetric CycleGAN [9] and Color CycleGAN [72]. They trained those architectures on both morphed Oulu-Casia and *ND-NIVL* [3] databases separately in the main translation direction from NIR to VIS spectrum. The CycleGAN, Pix2Pix and DenseUnet GAN outperformed the other architectures. The performance was thoroughly determined by multiple evaluation metrics that compared generated images with real images. The metrics were the following (as they are common metrics for image evaluation):

- Structural Similarity Index Measure (SSIM),
- Peak Signal to Noise Ratio (PSNR),
- Scale-Invariant Feature Transform (SIFT),

- Entropy (EN),
- color distance.

The best performing on both datasets was their DenseUnet GAN with the best achieving values in the majority of those metrics.

Wang *et al.* (2023) [70] proposed a solution for NIR-VIS translation for face images. They present their method named the Facial Feature Extractor (*FFE*) based CycleGAN (referred to as *FFE-CycleGAN*). The newly proposed architecture builds upon the original CycleGAN framework, introducing a new translate module, pixel-consistency loss and the FFE module mediated by domain invariant pre-trained DNN trained on large-scale dataset. Also, researchers collected a new *WHU VIS–NIR* dataset that has several advantages compared to the commonly used Oulu-Casia dataset. Namely, the face images are taken from various angles. This "makes the proposed network more stable for transferring face images" compared to the established Oulu-Casia dataset which only contains frontal face images. Since this study aimed to solve *face recognition* task on NIR spectrum images, the evaluation of the spectral translation was done by several face-recognition metrics. This new method was compared with CycleGAN, and Pix2Pix and performed the best followed by CycleGAN by a bigger margin. Also, demonstrated examples of generated images visually resembled ground truth noticeably more than the others. It should be noted, that the standard face recognition systems require consistency and precision of facial landmarks, therefore the NIR-VIS translation needs to be accurately translated, which is crucial for facial expression analysis as well. Therefore, the face-recognition metrics are relevant even for this use case.

Both above-mentioned studies utilize paired datasets. The other common methods employs CycleGAN as data can be unpaired [4].

1.4 Facial expression recognition

Facial emotion recognition (*FER*), also called *facial expression recognition* (as will be further used), is a topic in computer vision that aims to identify and understand human emotions based on facial expressions[4]. A person's emotional state can be described by basic emotions, which include happiness, sadness, anger, fear, surprise, and disgust (or additionally with neutral and contempt). Several studies employ this, other employ extended versions such as Compound Emotion introduced by Du *et al.* [10] or Russel's circumplex model of affect [54] (further discussed in 1.1.2).

Generally, FER can be divided by the methodology of a solution. First, *conventional approach*, extracts features from an image and then uses a classifier that delineates output (emotion type). The second approach is pure usage of DL methods such as CNN-based architectures. FER can also be divided by the usage of separate frames, or using the frame sequence throughout the video. Both the DL-based approach and sequence-based approach proved to be superior in performance compared to the conventional approach, although they have downfalls such as high computational complexity [31].

It is worth noting, that *micro expressions* (MEs) indicate more spontaneous and subtle facial movements that occur involuntarily. "They tend to reveal a person's genuine and underlying emotions within a short period of time" [31]. Therefore, if the system will be used on-fly, the inference time should be considered, so every micro-expression will be captured.

The existing solution employs *MobileNet* [58][21] architecture, which enables real-time inference since the MobileNet is not computationally demanding, however still well-performing, thus very efficient and useful in mobile devices. Current SOTA FER model *DDAMFN* [77] achieves on benchmark AffectNet (8 classes) $\sim 64.25\%$ accuracy, while the *POSTER++* [39] about $\sim 63.8\%$.

[4] Although other sensors can capture emotions than visual ones.

The first mentioned consists of two primary components: the Mixed Feature Network (MFN) that serves as the backbone and extracts resilient features using mixed-size kernels, and the Dual-Direction Attention Network (DDAN) that functions as the head and generates attention maps in two orientations. The second one employs both feature extraction and facial landmarks for classification and utilizes an attention mechanism as well. Other high-performing models are *Multi-task EfficientNet-B2* or *S2D*.

Note, that those networks are built for classification only. If the valence/arousal labels also want to be predicted, the network needs to be adjusted, such as proposed in [24].

1.5 Existing system

An already existing system [40] (2021) (later referred to as the *Original*) is designed for facial expression recognition from images in the visible spectrum. This solution, however, is lacking when insufficient illumination, such as at night. Therefore, there is a need for an illumination-independent solution.

The introduced system is designed for real-time usage and can recognize and classify facial expressions with up to five frames per second on the Raspberry Pi 4. Furthermore, it achieves near state-of-the-art performance on a significantly smaller network. To enhance scalability, the system was partitioned into two models within a single pipeline. The initial model enables facial detection, while the subsequent model discriminates the facial expression of a detected visage.

Two approaches were utilized for face detection - the Viola-Jones algorithm[5] [67][45] and a MobileNetV2-based *CenterFace* [73]. The first classifier had a faster inference time, however, did not achieve the same level of accuracy compared to the latter classifier.

The second part of the system, the facial expression recognition model, classified both categorical and dimensional expression predictors. One versatile system therefore predicted 8 categorical emotions[6] and valence/arousal predictors using box-like Russel's circuplex model of affect showed in figure 1.2. Regarding the architecture of the network, several models were tried including *EfficientNet-B0* [66] or *NASNetMobile* [82], however, the MobileNet-based [58][21] architectures outperformed others in accuracy and inference time.

Additionally, for testing and demonstrative purposes was developed a simple application that tracks faces and evaluates their facial expressions in real time.

1.6 Available Datasets

In this section will be introduced several datasets, that are relevant to this system. Namely datasets for NIR-VIS translation and facial expression analysis.

1.6.1 NIR-VIS face images datasets

There are several available datasets on the NIR-VIS translation of face images. Some of them [79][70] contain multiple emotions of each subject and some of them [79][70][3] are taken under laboratory supervision and are paired, whereas others in daily life and unpaired [75][36]. The following text describes the chosen available datasets.

- **Oulu-CASIA NIR&VIS face expression database**
  The commonly used dataset [79] has 80 subjects with 6 basic emotions per subject (480 altogether). Each image is paired with its counterpart and all images were taken under 3 types of illumination. Its extended morphed3 version contains a total of 11,120 images for

[5] It is also known as the Haar cascade classifier.
[6] neutral, angry, sad, happy, surprise, disgust, fear and contempt

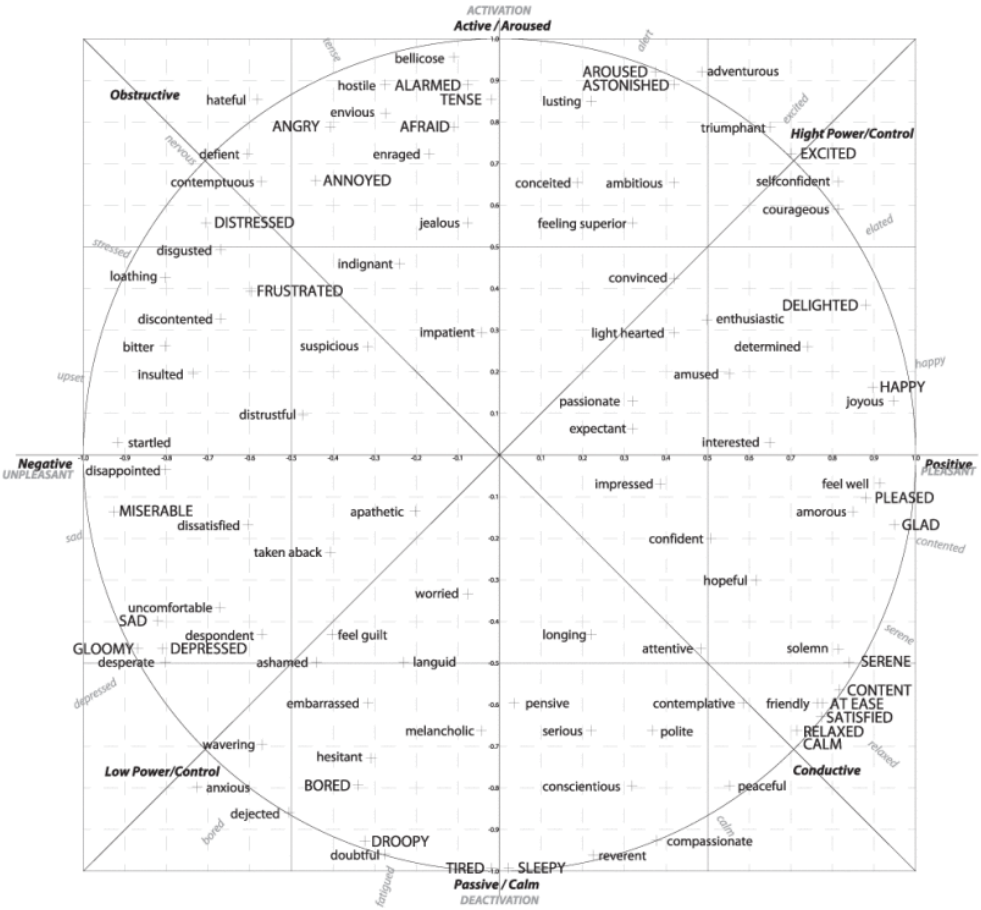


Figure 1.2 Figure demonstrates the circumplex model of affect for both valence and arousal labels, which has been used in the existing solution.

each illumination and each spectrum. The dataset contains both 320x240 raw images and preprocessed grayscale images (cropped and aligned by eyes) 128x128. The dataset is taken under laboratory conditions. Half of the subjects are caucasian race and the other half of mongoloid race.

- **CASIA NIR-VIS 2.0 Face Database**
  The CASIA NIR-VIS 2.0 [36] face database contains 17,580 images of 725 subjects. VIS and NIR face images of each subject range in quantity from 1 to 22 and 5 to 50, respectively. The dataset contains raw images (640x480) and preprocessed images (cropped and aligned by eyes to 128x128). The CASIA 2.0 comprises images of large diversities in illumination, expression, distance, and pose. Each image is randomly captured, so the NIR and VIS images of one person are unpaired.

- **ND-NIVL dataset**
  Bernhard *et al.* [3] provided a paired dataset with 24605 images of 574 subjects. Of the 574 subjects 402 appear in multiple sessions. "The NIR images have a resolution of 4770×3177 and the visible light images have a resolution of 4288×2848". However, the NIR-VIS images were not acquired simultaneously.

- **WHU VIS–NIR paired face dataset**
  A newly collected paired database by Wang *et al.* [70] (2023) is captured synchronously by a binocular camera[7] in normal illumination. Dataset emphasises different angles of a face in daily life and randomness in expression – "neutral-frontal, tilt-up, tilt-down, left-rotation, right-rotation, blank and smile". It contains 80 subjects with a total of 12,800 images of NIR and VIS.

- **The BUAA-VisNir face database**
  The BUAA-VisNir face database (Huang et al. 2012) was developed in 2012 and includes 150 subjects with 40 images per subject. For each subject, there are 9 NIR-VIS image pairs at a resolution of 640×480 pixels and 287x287 is a cropped publicly available version. Paired images were captured simultaneously using a multispectral imaging device[8] and the VIS images are in grayscale. All images of every subject are collected under 9 poses and expressions (including neutral-frontal, left-rotation, right-rotation, tilt-up, tilt-down, happiness, anger, sorrow and surprise). However, this database has only faces of mongoloid race.

- **LAMP-HQ dataset**
  The LAMP-HQ [75] is a heterogeneous high-resolution face dataset, containing 56,788 NIR and 16,828 VIS images of 573 subjects with large diversities in pose, illumination, attribute, scene, and accessory. This large and diverse dataset is, however, unpaired.

A brief summarization of the datasets is captured in the table 1.1.

1.6.2 Facial Expression datasets

The most notable and used VL datasets are the following ones that were utilized in above mentioned studies for facial expression analysis [40][4].

- **AffectNet**
  AffectNet [42] is a large facial expression dataset with around 0.4 million images manually labelled for the presence of eight (neutral, happy, angry, sad, fear, surprise, disgust, contempt) facial expressions along with the intensity of valence and arousal. Furthermore, the database

[7] Inter-ocular distance of the acquired imagery is about 20 mm.
[8] Captured with one lens.

name	resolution	# NIR/VIS	lab	paired	diversity
Oulu-CASIA	320x240	5,560/5,560	yes	**yes***	7 emotions
Oulu-CASIA 2	640x480	total 17,580	yes	no	wide
ND-NIVL	∼ 4K	22,264/2,341	yes	**yes**	
WHU VIS–NIR	∼ FullHD	6,400/6,400	yes	**yes***	rotations + 2 emotions
BUAA-VisNir	640×480; cropped 287x287	1950/1950	yes	**yes****	rotations + 5 emotions
LAMP-HQ	wide	56,788/16,828	no	no	wide

Table 1.1 This table depicts attributes of NIR-VIS datasets. A single asterisk in a paired column means paired photos acquired simultaneously with a binocular camera, two asterisks then with a one-lens multispectral camera.

consists of additional not labelled images. All images were downloaded from the Internet, therefore the images are from a wild environment. However, the database is hugely imbalanced classes with the *happy* and *neutral* classes having majority 75% of data, while the rest is between $2 - 10$ [40].

- **MorphSet**
  MorphSet [68] is an augmentation framework that employs facial morphing techniques to expand categorical emotion datasets, effectively encompassing a broad spectrum of valence-arousal levels. This dataset was generated by merging three categorical datasets widely utilized in psychology [33][13][44], resulting in approximately 300,000 images of posed facial expressions with associated dimensional valence-arousal labels acquired within a laboratory environment[4].

There are other databases such as [15][35][32][33] thoroughly compared in [40]. However, each has drawbacks such as missing valence-arousal labels, low resolution or inappropriate settings.

Chapter 2
Computer Vision

This chapter provides an introduction to Deep Learning (DL) with a focus on Computer Vision. It will cover the basic principles and present several neural network architectures used in this thesis.

Artificial Intelligence (AI) is a broad and interdisciplinary field in computer science that focuses on creating intelligent systems capable of performing tasks that typically require human intelligence. At the heart of AI lies *Machine Learning* (ML), its subset, that leverages algorithms and statistical models that can learn on provided data to perform certain tasks such as prediction without being explicitly programmed. *Deep Learning* (DL) is a specialized branch of ML that harnesses the power of *Artificial Neural Networks* (ANNs), specifically *Deep Neural Networks* (DNN).

After the AI is described, we can introduce *Computer Vision*, a field of AI that focuses on enabling computers to interpret and understand visual information from the world, including images and videos. This area plays a critical role in various tasks, such as object recognition, image classification or facial expression recognition and is employed in a wide range of applications, including manufacturing, healthcare, transportation, agriculture or retail. The computer vision system is tackled by a conventional approach or approach using DL. The conventional approach tends to be more computationally feasible, however less accurate. The increasing employment of DL stems from the growing affordability and computational power of modern computers. The most common type of ANN that solves Computer Vision tasks is *Convolutional Neural Network* (CNN). However, before we explore the CNNs, it is essential to introduce *feedforward* ANNs and their fundamental principles. These principles serve as the foundation upon which ANN's various subtypes, such as CNN, are constructed.

2.1 Feedforward Artificial Neural Network Introduction

ANNs have a rich history dating back to the mid-20th century, and it has continued to evolve. A significant surge in the development of ANNs occurred during the late 20th century when there was a substantial improvement in computational resources and the availability of vast amounts of data for research. It was during this period that researchers were able to explore and study ANNs extensively.

In the early 21st century, especially the latest decade, ANNs made remarkable progress, becoming capable of addressing a myriad of tasks, including classification, prediction, text and image generation, translation, and many others. DL found applications in a wide range of fields, including manufacturing industries, healthcare, financing, entertainment, robotics and much more [55].

As DL progressed, a multitude of new, distinct, and complex network architectures emerged. Nowadays, each of the tasks mentioned above has its specialized architecture, which may differ significantly from the original ANN introduced many years ago. However, it is important to note that the original ANNs remain fundamental in understanding the principles of Deep Learning. Therefore, before delving into the intricacies of various architectures, it is essential to introduce the foundational structure of ANN.

2.1.1 Artificial Neurons and its arrangement

The first introduction of ANNs was in 1958 the *Perceptron* [52], a single computational unit, inspired by biological neurons, that resembles nowadays used units, aptly called a neuron[1]. A neuron can be defined as a function

$$z(\mathbf{x}, \mathbf{w}) = f(\sum_{i=0}^{n} x_i w_i) \tag{2.1}$$

where $\mathbf{x} = (x_0, x_1, ..., x_n)$ and $\mathbf{w} = (w_0, w_1, ..., w_n)$[2] which are the input numbers and the corresponding weights to those inputs respectively, thus both having the same length n. As depicted in the formula, input numbers will be multiplied by their weights and summed. The final product is then applied to a nonlinear function, also called *activation function*. This is one of the key principles of ANN and will be further discussed in 2.1.6.

The original Perceptron used one such unit that received input and the utilized weights[3] (here as $\mathbf{x}$ and $\mathbf{w}$ respectively), which were optimized by the so-called *backpropagation* (a part of a training process discussed later), to produce an output number – a prediction. Additionally, a special constant node was added to the input, $x_0 = 1$ (with corresponding weight w_0)[4], referred to as *bias*, which should increase the network's capabilities. For the activation function was used a *step function*, which is nowadays not used.

This idea evolved into *Multilayer Perceptron* (MLP), which stacked multiple Perceptrons in one layer. Each perceptron was connected to the input, as was the single one Perceptron, and together they formed so-called *hidden layer*. Since there were multiple Perceptrons, more outcomes arose and therefore an *output layer* was created, connecting the Perceptrons in a hidden layer with each output node, also a Perceptron-like. The number of nodes would vary in the hidden or output layer as well as the number of hidden layers. Those types of layers as in MLP, are nowadays referred to as *fully connected layers*. The exemplary MLP is depicted in figure 2.1. Also, since the MLP utilizes hidden layers, it is labelled as a Deep Neural Network (DNN), hence it is a Deep Learning tool. As the development progressed, multiple architectural improvements emerged such as *Dropout*, *Batch Normalization* or different activation functions and nowadays are understood as a feedforward neural networks (FNN), a subtype of ANNs.

To clarify, the whole process of forwarding the input through the network makes the desirable outcome, when the weights are optimized.

2.1.2 Training Process

A training process is a key element in the ANNs because it enables the network to learn on provided data. The process could be divided into forward pass which calculates the output based on input data, error calculation quantifying the correctness of an input and finally backward

[1] In the introduced Perceptron was the single neuron called a Perceptron, as well as in Multilayer Perceptron discussed later. Therefore, the Perceptron is a type of neuron.

[2] Both weights and input numbers are floating points.

[3] Including the bias.

[4] Further will be bias and its weight included in the weights and input to facilitate understanding.

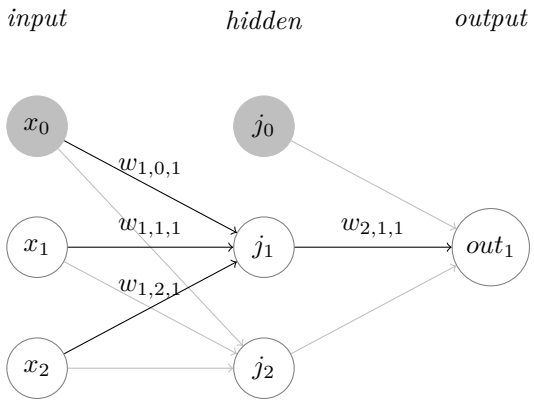


Figure 2.1 Multilayer Perceptron Network with input $\mathbf{x}$, one hidden layer $\mathbf{j}$, single output out_1 and corresponding set of weights 3. The bias is denoted as grey-filled neurons.

pass, the process of parameter optimization. In the backward pass, the calculated error by a *loss function* [5] is employed as a metric that the network aims to minimize.

This training process is done for all the input data and encloses the whole, referred to as *epoch*. Typically, dozens of epochs need to be iterated to tune the network to plausible results. The training process ends when a loss function of a network converges and no notable improvements emerge. Additionally, in the training process is an optionally (though commonly) employed batch-based approach, which samples data into sets, then forwards them into the network and afterwards backwards them. That is one of the strategies of optimizers trying to enhance the process. Overall, the whole training of one epoch is demonstrated in figure 2.3.

When training an ANN, it is crucial to set up the correct hyperparameters, parameters of the network set-up, to ensure convergence. Also, the process called *overfitting* needs to be evaded. Overfitting can be defined as fitting (a strive to minimize the loss) a provided set of data, instead of trying to capture and learn a general essence of the target task, referred to as *generalization*.

To prevent it, we can add some *regularization* techniques such as *Dropout*, *L1* or *L2 regularization*. But, firstly we need to be able to detect the phenomenon. The main indicator of overfitting is a *validation error* of so-called *validation split*. Before the training begins, the available data is split randomly[6] into parts – a *training set* and a *validation set*. The firstly mentioned split is, as the name suggests, employed for training the network. The second split is then used to validate if the network has generalization capabilities and if the loss function similarly declines as the training loss. If, on the other hand, validation loss increases from some point, it is a sign of overfitting. That is demonstrated in figure 2.2 with a highlighted *sweet spot*, being the place between overfitting and underfitting (a state when the network is not learned enough yet).

Additionally, the data needs to be split into the third part – a *test set*. This set should be evaluated when the development of the network ends, which will show true generalization capabilities. It needs to be done because we can overfit the validation data with imprudent tuning of hyperparameters or network architecture.

2.1.3 Loss Functions

As mentioned earlier, the loss function, sometimes referred to as the cost function when applied to a batch of samples, quantifies the error between the predicted values and the target values (the correct outputs). Subsequently, during the backward pass, the loss function is employed to

[5] Generally, it is called the *objective function*. The loss function is for minimization (also *cost function*) and *fitness function* is for maximization. We will, for simplification, consider only minimization, thus loss function.

[6] When tasks such as classification, the splits need to be class-wise well-balanced as well as mini-batches.

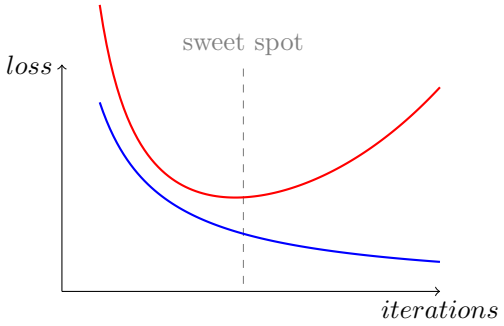


Figure 2.2 Depiction of overfitting with the blue curve being the training loss and the red curve being the validation loss. A sweet spot is also highlighted

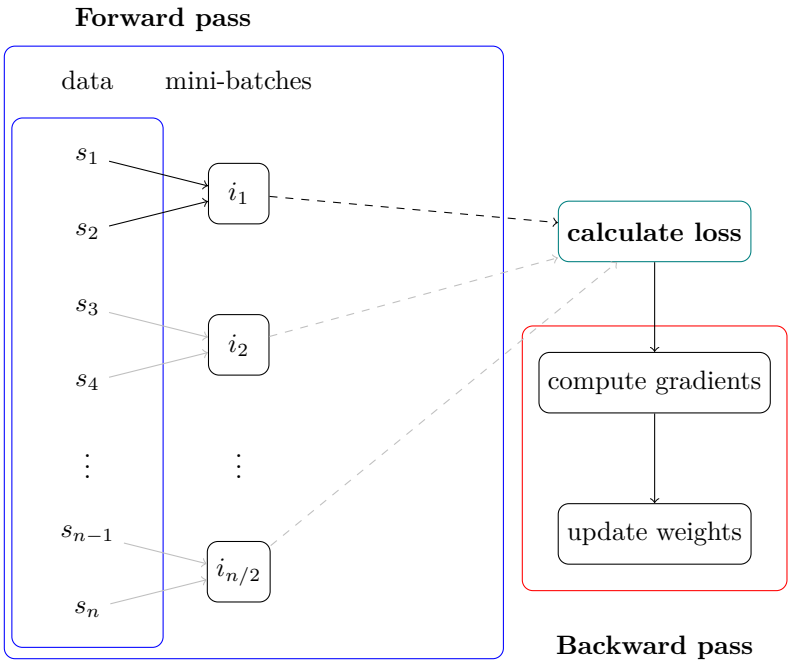


Figure 2.3 Training process utilizing mini-batch with a highlighted (by black arrows) forwarding and backwarding of a first mini-batch. A dashed connection represents a feed forwarding through the network itself.

adjust the network parameters to minimize the overall loss. There is a variety of loss functions, that behave on their inputs differently, thus they affect the learning process. Therefore, it is crucial to choose a suitable function for each task and the data's nature. The following functions are the most common ones and they are the ones that are used in this thesis.

2.1.3.1 Mean Squared Error (MSE)

MSE is one of the most common ones for tasks, that are different from classification such as regression. It is also called *L2 loss* and it has other variants such as *Root Mean Squared Error* (RMSE). The definition is as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.2}$$

where y_i and $\hat{y}_i$ represents the target and predicted value respectively for the i-th data samples of a n-sized set.

2.1.3.2 Mean Absolute Error (MAE)

MAE, also known as *L1 loss*, is also used in regression tasks and others. Unlike MSE, which squares the differences and gives more weight to large errors, MAE treats all errors equally, making it robust to outliers. However, it is less sensible to small errors compared to MSE. As the name suggests, the definition is:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.3}$$

with the same interpretation of variables as in MSE (2.2).

2.1.3.3 Cross-Entropy Loss (Binary and Categorical)

Cross-entropy loss is commonly used in classification tasks. There are two variations: binary cross-entropy for binary classification and categorical cross-entropy for multiclass classification.

For binary cross-entropy, the formula is:

$$\text{Binary Cross-Entropy} = -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right) \tag{2.4}$$

where n is the number of data points. y_i is the true class label (0 or 1) for the i-th data point and $\hat{y}_i$ is the predicted probability of the data point belonging to class 1.

For categorical cross-entropy, when dealing with multiple classes, the formula is:

$$\text{Categorical Cross-Entropy} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) \tag{2.5}$$

where similarly n is the number of data points. C is the number of classes, y_{ij} is an indicator variable (1 if the i-th data point belongs to class j, 0 otherwise) and $\hat{y}_{ij}$ is the predicted probability of the i-th data point belonging to class j.

2.1.4 Backward pass

The backward pass is the process where the network "learns", which is achieved by updating all the learnable parameters, weights and biases, to the values that minimize the loss.

First, the weights are initialized for which is commonly employed *Xavier (Glorot) initialization* [12], that is suitable for activation functions with mean around 0 (such as *sigmoid* or *hyperbolic tangent*), or *He initialization* [18], which is more suitable for *ReLU*. Similarly, the biases are also initialized, commonly to 0.

There are many strategies for how to do a backward pass using so-called optimizers, which will be introduced as follows. Each consists of calculating the gradients (*backpropagation*) and subsequently updating the weights. The whole process of training is depicted in figure 2.3.

2.1.5 Optimizers

2.1.5.1 (Stochastic) Gradient Descent

The fundamental *Gradient Descent* algorithm serves as the foundation for numerous optimizers, making it a solid starting point for understanding other optimization techniques. It utilizes the idea of gradient, which shows the steepest growth of a function from a current position. In the case of minimizing the objective function, the steepest decline is then in the opposite direction of a gradient. With this approach we can optimize all the variables (parameters) utilizing a *step*, that moves to a more promising position in a variable space. An update of weights is defined as follows

$$\theta^{(i+1)} = \theta^{(i)} - \eta \nabla \mathcal{L}(\theta^{(i)}) \tag{2.6}$$

where $\theta^{(i)}$ represents the model parameters at the i-th step, η is the *learning rate* – the defined size of the step, usually has a value around 0.001. Nevertheless, $\nabla \mathcal{L}(\theta^{(i)})$ is the gradient of the loss function $\mathcal{L}$ with respect to the parameters $\theta^{(i)}$. The update for the one specific j-th weight is

$$w_j^{(i+1)} = w_j^{(i)} - \eta \frac{\partial \mathcal{L}}{\partial w_j}(\theta^{(i)}) \tag{2.7}$$

where $\frac{\partial \mathcal{L}}{\partial w_j}(\theta^{(i)})$ is the partial derivative of the loss function with respect to the parameter w_j at the parameter values $\theta^{(i)}$.

The whole process of backpropagating is then applied to previous layers with the usage of the *chain rule* of calculus, which enables the computation of gradients for the earlier layers in the neural network.

The Gradient Descent algorithm is reliable, though slow, as each data sample needs to be forwarded for computing one parameter update and in big amounts of data, it is computationally unfeasible. In this case, it will come in handy to employ faster *Stochastic Gradient Descent* (SGD) [53] that randomly picks samples of data in so-called *mini-batches* [7].

This significantly decreases computational time, however, the loss function is only computed on a subset of data, which makes it an estimation. The higher the mini-batch size, the more accurate the estimation, however slower, so a reasonable size of the mini-batch might be for example 256. Also, it is necessary to create class-wise balanced mini-batches, but still random. After the loss is estimated, the gradients can be computed and end and iteration with the update of weights.

[7] Original SGD uses mini-batch of size 1, although it is more practical to use batches because it makes the loss estimation more stable.

There are also variants of SGD such as *SGD with momentum*, that is generally better than the original SGD. The addition of momentum, which means that a current gradient is corrected by previous gradients reduces oscillation and therefore accelerates the convergence.

2.1.5.2 Adam

The *Adaptive Moment Estimation* (Adam) optimizer [30] is a popular optimization algorithm, used in this work as well, that builds upon the principles of the Gradient Descent algorithm. It combines the benefits of both the *Adagrad* and *RMSProp* optimizers to offer efficient and adaptive learning rates. Specifically by utilizing both the first moment (mean) and the second moment (uncentered variance) of the gradients to adaptively adjust learning rates for individual parameters.

Adam has many advantages such as that the step size of Adam update rule is invariant to the magnitude of the gradient, which helps a lot when going through areas with tiny gradients (such as saddle points or ravines). In these areas, SGD struggles to quickly navigate through them. Also, at the beginning, it optimizes very fast.

However, later research showed that Adam struggles to find the optimal solution, therefore some state-of-the-art solutions on some tasks use SGD with momentum [71].

As a response to those disadvantages are variants such as *Nadam* or optimizer *SWATS*, which is a simple switch in between the training from Adam, that quickly decreases the loss function, to the SGD with momentum, which is better in finding the optimal solution.

The use of the Adam optimizer is prevalent in DL and has contributed to the success of many state-of-the-art models across various domains.

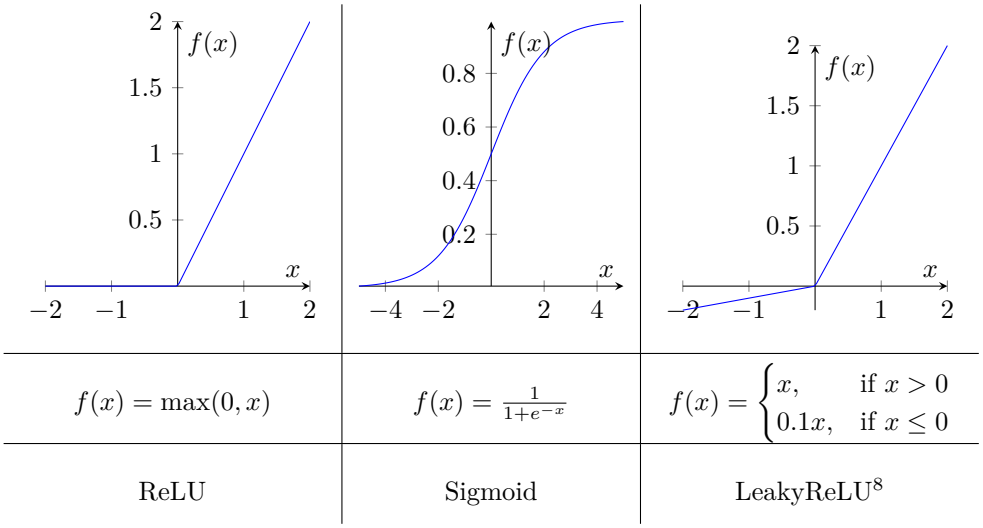
2.1.6 Activation Functions

Activation functions are a vital component of neural networks. They introduce non-linearity into the network, allowing it to learn complex relationships in data. Each layer of a neural network typically applies an activation function to its input, transforming it before passing it to the next layer as in equation 2.1. The choice of activation function can significantly impact a model's training and performance. Here are activation functions, that are used in this thesis, and happens to be commonly used as well:

- **Sigmoid Function**: The sigmoid function, also known as the logistic function, maps input values to a range between 0 and 1. It is commonly used for models where we have to predict the probability as an output. However, suffers on so-called *vanishing gradient / exploding gradient*, hence not frequently used in hidden layers.
- **Rectified Linear Unit (ReLU)**: The ReLU activation function is one of the most widely used functions in hidden layers as it is not prone to vanishing/exploding gradient problems. Although, it suffers from *dying ReLU* problem.
- **Leaky Rectified Linear Unit (Leaky ReLU)**: The Leaky ReLU is a modification of the standard ReLU activation function and attempts to address the dying ReLU problem. While ReLU sets all negative input values to zero, causing neurons to "die" and not update their weights during training, Leaky ReLU allows a small, non-zero gradient for negative inputs.
- **Softmax Function**: The softmax function is often used in multi-class classification tasks. It transforms a vector of raw scores into a probability distribution over multiple classes, making it suitable for output layers in such tasks, and thus is easily interpretable for a human.

The first three activation functions are plotted in a table 2.1 with their definitions. Also, another notable activation function is the Hyperbolic Tangent function (tanh) which is also used.

Table 2.1 Comparison of Activation Functions: ReLU, Sigmoid, and Leaky ReLU.



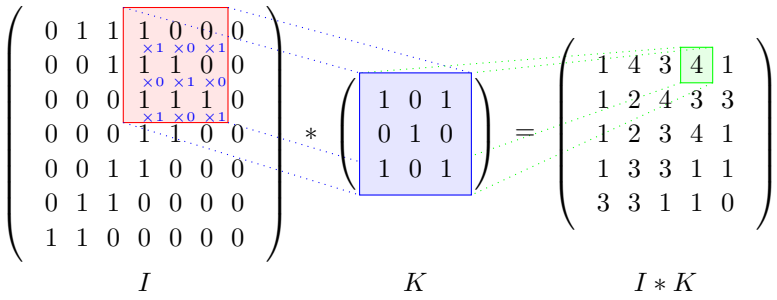


Figure 2.4 Figure demonstrates process of convolution between the input image I and filter K, where the output is $I * K$ on the right part of figure.

2.2 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are at the heart of numerous computer vision tasks, such as image detection and classification, among others. They have witnessed significant advancements, particularly during the early 2010s, with ResNet [19] being a pivotal milestone in their development. CNNs, in whole or in part, find widespread application in this work. One significant distinction between CNNs and FNNs is the presence of convolutional layers preceding the fully connected layers.

2.2.1 Convolution

Convolution is a mathematical operation used in various fields, including signal processing and image processing. It involves combining two functions to produce a third function that expresses how one function modifies the other. In the context of image processing, convolution is often used to extract features or apply filters (also kernels) to an image.

The discrete 2D convolution operation, commonly denoted as $(f * g)(x, y)$, between an image $f(x, y)$ and a filter $g(x, y)$ is defined as:

$$(f * g)(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(x - m, y - n) \cdot g(m, n) \tag{2.8}$$

where $(f * g)(x, y)$ represents the result of the convolution at position (x, y). $f(x, y)$ is the pixel value at position (x, y) in the input image and similarly $g(x, y)$ is the value of the filter at position (x, y). The summation is performed over all possible values of m and n, considering the entire filter and image. Thus, for a filter of dimensions 3×3 would be possible m and n values -1 and 1 respectively.

The convolution operation involves sliding the filter over the image, calculating the element-wise product of the filter and the image at each position, and then summing these products to obtain the output of the operation. The process is captured in figure 2.4.

Since the convolution itself decreases the dimensions of the input image, we can, if desired, preserve the dimensions by increasing the input image on its borders before convolution. This commonly used enlargement is called *padding* and has many variants of how to fill new cells, such as *zero padding* filling with zeros, or *mirroring*.

In convolution is also commonly used *stride* – the size of displacement, or *dilation*.

2.2.2 Architecture

The architecture of CNN could be divided into 2 parts – a *feature extraction* part and subsequent Feedforward Neural Network. The first part ensures the extraction of characteristics (features) of the image, whereas the second one does the classification (or other desired task) of those features. The feature extraction part, a core element of CNN, has on input the original image that is forwarded into the following convolutional layers with decreasing dimensions.

Between each convolution, an activation function is applied, and a process called *pooling* is performed. Pooling involves sliding a small box (typically a square of 2×2 dimensions which can also have a different stride size, as explained in 2.2.1) across the output of the convolution, resulting in the averaging or selection of the maximal value. Therefore, this operation reduces the dimensions by double (if *stride* $= 1$ and dimensions 2×2). With decreasing dimensions is increasing the amount of so-called *channels*.

This architecture of the feature extraction part propels each layer of convolution to capture differently detailed features of an image. The first layers capture big, general characteristics, whereas layers at the end capture small details. Additionally, it is important to mention the concept of the receptive field, which refers to the area of the input that affects a convolution's output. Receptive fields can be enlarged using *dilated convolutions*, which allow the network to capture information from a broader context. This is especially valuable for recognizing larger patterns and structures within the input data.

Finally, the output of the last convolutional layer is flattened and forwarded into fully connected layers, which classify the input image. The whole architecture is depicted in figure 2.5.

It is worth mentioning that the filter's values (weights) are parameters to learn, and the filter values are shared for each position on the image. This idea enhances the performance of the network and significantly decreases the amount of learnable parameters, hence allowing the network to go deeper.

2.2.3 Types of CNN

Nowadays, there are many types of CNNs such as *Faster R-CNN* [50] that detect and classify objects, *MobileNet* [21] which is specially designed for mobile devices and also is used in an existing solution of the facial expression detection, or *VGG* [63] family of networks and many more.

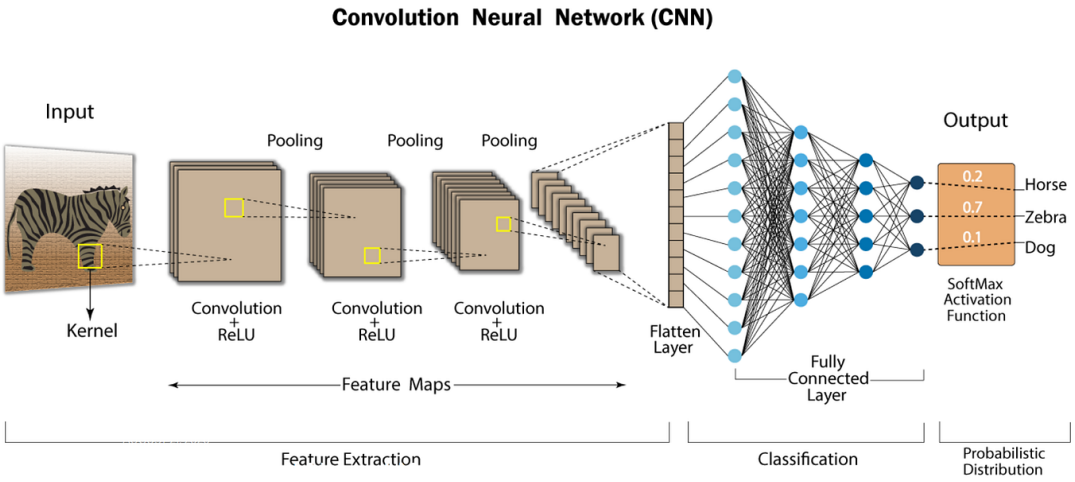


Figure 2.5 The figure illustrates the architecture of a CNN.

However, the following text will describe the core principles of networks ResNet and DenseNet, since their principles are utilized in this work.

2.2.3.1 ResNet

Residual Networks (ResNet) was introduced by He et. al. [19] in 2015. Its innovative idea of *Residual Blocks* incorporated into classical CNN made a big impact on Computer Vision and inspired other architectures. It has many variants such as *ResNet-34*, *ResNet-50* or *ResNet-101* differing in their depth.

The residual block is built on the idea that neural networks can more effectively learn the difference (residual) between the current input and the desired output, rather than trying to learn the entire mapping from input to output directly. This concept simplifies training and allows the creation of extremely deep networks by creating shortcut paths for gradient flow. Each residual block consists of a few convolutions followed by the addition of the block's input to the output of the block. This can be seen in subfigure *(a)* in figure 2.6.

The ResNet's architecture is then a concatenation of those residual blocks followed by usually one fully connected layer.

2.2.3.2 DenseNet

DenseNet [23] builds on the idea of residual blocks and enhances it by creating *dense blocks*, where after each convolution is forwarded its output to all the following convolutions. Also, as opposed to the residual blocks, dense block uses concatenation instead of addition. Combining with the residual emerges a *residual dense block*, which is used in similar architectures.

Dense block is depicted in comparison with a residual block in figure 2.6.

2.3 CycleGAN

Cycle-Consistent Generative Adversarial Network (CycleGAN) [81], introduced in 2017, is a type of GAN [14], which is designed for unpaired image-to-image translation tasks. Typically is used in domain-to-domain translation tasks, such as horse-zebra translation (as in the original paper) or colourization. The key principle in this network is a *cycle-consistency*, which will be discussed further.

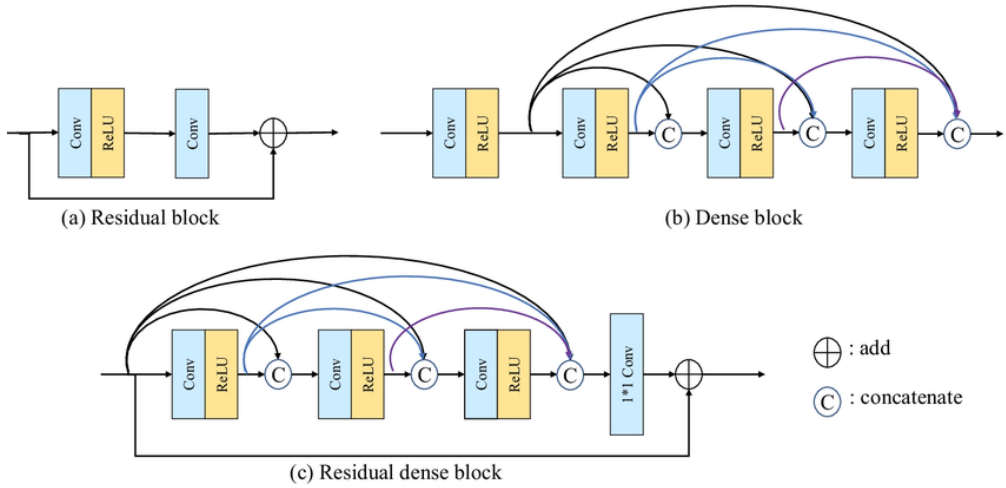


Figure 2.6 The comparison of *residual block (a)*, *dense block (b)* and *residual dense block (c)* [34].

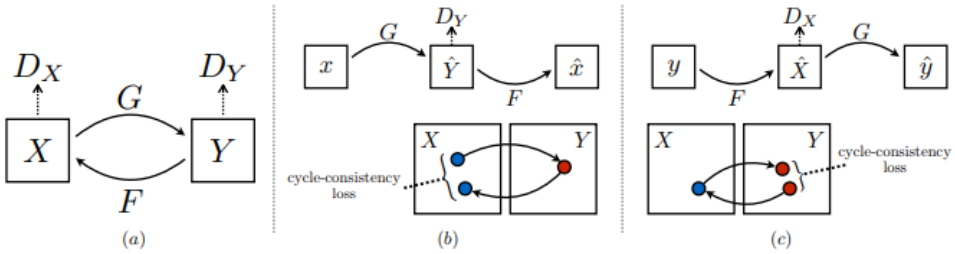


Figure 2.7 Subfigure *(a)* demonstrates CycleGAN architecture with X and Y being the domains, G and F being the generators and D_X with D_Y being the discriminators. Both subfigures *(b)* and *(c)* demonstrate the cycle-consistency loss for both generators. Also, there is captured, in which phase is calculated adversarial loss by the discriminators.

2.3.1 Objective

The primary objective of CycleGAN is to learn a mapping between two domains, typically referred to as the source domain and the target domain, here defined as X and Y respectively. Given a set of images from both domains, the network aims to generate images in a way that they appear as if they belong to the target domain.

2.3.2 Architecture

CycleGAN consists of two generators and two discriminators:

- **Generators:** These are responsible for transforming images from one domain to the other. In a typical CycleGAN setup, one generator converts source domain images into target domain style, while the other generator performs the reverse operation. The architecture of a generator can be various where the commonly used is UNet-based generators or ResNet-based ones.
- **Discriminators:** These networks are responsible for distinguishing between real and generated images. The two discriminators are used for the source and target domains, ensuring

that the generated images are realistic and indistinguishable from real images in both domains. The architecture for the discriminator commonly used is the one used in a PatchGAN [26], UNetGAN [59] or in the original GAN [14].

The generators and discriminators are depicted in subfigure *(a)* of figure 2.7 with G and F being the generators transforming an image to domain Y and X respectively. Similarly, the discriminators are D_X and D_Y discriminating the images from domain X and Y respectively.

2.3.3 Loss Functions

In CycleGAN, three main loss functions are used for training:

1. **Adversarial Loss ($\mathcal{L}_{\text{adv}}$):** This loss encourages the generators to produce images that are indistinguishable from real images. For the generator G transforming images from source to target domain and its corresponding discriminator D_{Y}, the adversarial loss generator G and its discriminator D_Y is defined as:

$$\mathcal{L}_{\text{adv}}(G, D_{\text{Y}}, X, Y) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D_{\text{Y}}(x)] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log(1 - D_{\text{Y}}(G(y)))] \tag{2.9}$$

   G aims to minimize this objective against an adversary D_Y that tries to maximize it, i.e., $\min_G \max_{D_Y} \mathcal{L}_{\text{adv}}(F, D_{\text{X},X,Y})$ Similarly, there is a loss for generator F and adversary discriminator D_X: $\mathcal{L}_{\text{adv}}(F, D_{\text{X}}, X, Y)$.

2. **Cycle-Consistency Loss ($\mathcal{L}_{\text{cyc}}$):** The cycle-consistency loss, the key innovation, enforces the cycle-consistency property, which generates a false image and afterwards reconstructs a false image back to the source domain, which should be identical with the source image. For the generators G and F that map images between domains, it is defined as:

$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[||F(G(x)) - x||_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[||G(F(y)) - y||_1] \tag{2.10}$$

3. **Identity Loss ($\mathcal{L}_{\text{iden}}$):** The identity loss ensures that the generators preserve the content of an image when translating within the same domain. It is used to minimize the difference between the generator's output and the original input:

$$\mathcal{L}_{\text{iden}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[||G(x) - x||_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[||F(y) - y||_1] \tag{2.11}$$

Overall, the adversarial loss encourages the generators to produce realistic images, while the cycle-consistency loss enforces the cycle-consistency property. The identity loss helps to maintain the content of the original image during translation, however, it is not necessary to use it.

All the losses above are then connected to the total loss for a generator. Also, because the partial loss functions need to be in balance, each one has its multiplicative constant tuned for the problem domain and the data. The loss for the discriminator then attempts to maximize the adversarial loss of the adverse generator. And for this purpose is commonly used L1 loss (defined in 2.3)

2.4 MobileNet Architecture

MobileNet is a class of efficient models [21][58][20] for mobile and embedded vision applications capable of running on CPUs thanks to less than 5 million parameters. There are multiple available versions: *V1*, *V2* and *V3* which all are based on a streamlined architecture that uses depth-wise separable convolutions to build lightweight deep neural networks. MobileNet is widely used in many real-world applications including object detection, fine-grained classifications, face attributes, and localization.

2.4.1 Depthwise Separable Convolution

In traditional convolutional layers, each filter is applied across the entire depth of the input. In contrast, depthwise separable convolutions divide this operation into two parts: a depthwise convolution and a pointwise convolution. The depthwise convolution applies a single filter per input channel which filters the input image, and the pointwise convolution then applies a 1x1 convolution to combine the outputs of the depthwise convolution. The benefit of using this convolution is that it decreases the number of computations without significantly decreasing the performance – standard convolution has 9 times more multiplications than that of the Depthwise separable convolution when using output channels of convolution as 1024 and filter size as 3 [64]. A comparison of all convolutions is depicted in figure 2.8.

2.4.2 Network Structure

The MobileNet structure is built on depthwise separable convolutions as mentioned in the previous section. The model takes the form of a base architecture followed by a simple fully connected layer. The base architecture is then followed by an average pooling layer and a fully connected layer for classification.

2.4.3 Width and Resolution Multipliers

The width multiplier provides a trade-off between computational cost and classification accuracy. It reduces the number of input and output channels proportionally by introducing the global hyperparameter $\alpha \in (0, 1)$.

The resolution multiplier is another hyperparameter that modifies the input image resolution. It allows the model to be adaptable to different computational resource constraints.

2.4.4 MobileNet V2 & V3

MobileNetV2 introduces Linear Bottlenecks and Inverted Residual Blocks to preserve important information and reduce parameters, making the network faster. MobileNetV3 optimizes V2 with an automated system called Neural Architecture Search (*NAS*) and a trimming algorithm called NetAdapt. Both versions are designed for devices with limited computational power, balancing latency, accuracy, and size.

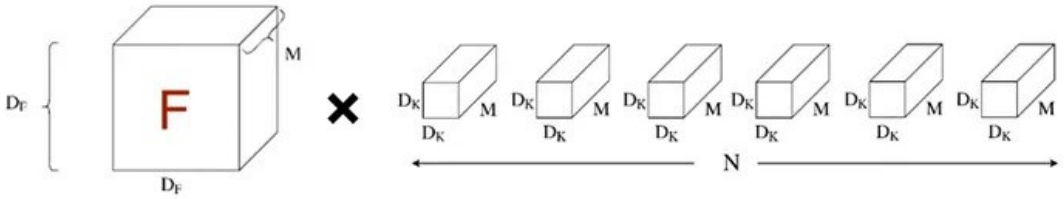
2.5 DDAMFN

The *Dual-Direction Attention Mixed Feature Network* (DDAMFN)[80] is a novel number one state-of-the-art network architecture specifically designed for *Facial Expression Recognition* (FER). The DDAMFN consists of two primary components: the *Mixed Feature Network* (MFN) and the *Dual-Direction Attention Network* (DDAN).

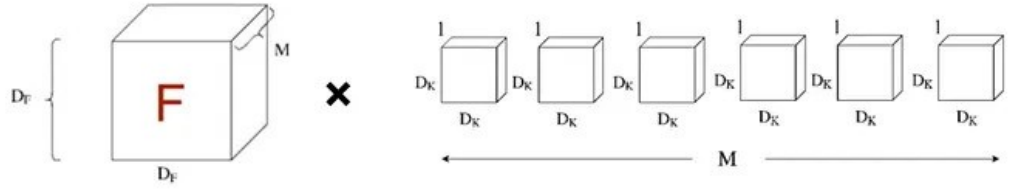
The MFN serves as the backbone of the network. It enhances the network's capability by extracting resilient features using mixed-size kernels. This approach allows the network to capture a wide range of facial features, which contributes to its robustness.

The DDAN functions as the head of the network. It introduces a new *Dual-Direction Attention* (DDA) head that generates attention maps in two orientations. This enables the model to capture long-range dependencies effectively, which is crucial for recognizing complex facial expressions.

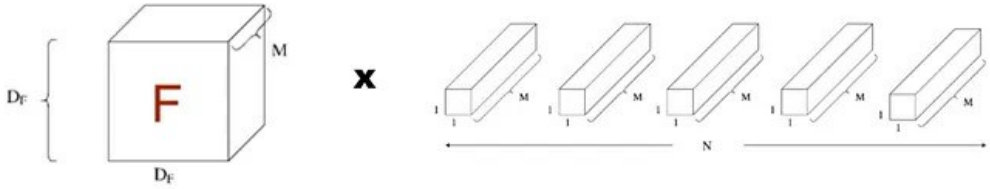
To further improve the accuracy, a novel attention loss mechanism for the DDAN is introduced. This mechanism ensures that different heads focus on distinct areas of the input. By doing so, the network can pay more attention to the most informative parts of the facial images.



(a) Standard convolution



(b) Depthwise convolution



(c) Pointwise convolution

Figure 2.8 Comparison of standard, depthwise and pointwise convolution [64], where N is #input channels, M is #output channels, D_F is input image size and D_K is kernel size.

Experimental evaluations on several widely used public datasets, including AffectNet, RAF-DB, and FERPlus, demonstrate the superiority of the DDAMFN compared to other existing models. These results establish the DDAMFN as a state-of-the-art model in the field of FER with accuracy on AffectNet8[9] 64.7%.

In summary, the DDAMFN is a robust and lightweight network architecture for FER. It leverages mixed-size kernels and dual-direction attention to extract resilient features and capture long-range dependencies. Its novel attention loss mechanism further enhances its performance, making it a leading model in the field.

This robust and lightweight network leverages its architecture depicted in figure 2.9, however, it is computationally demanding, since it uses attention mechanism.

[9]Denotes AffecNet with 8 classes.

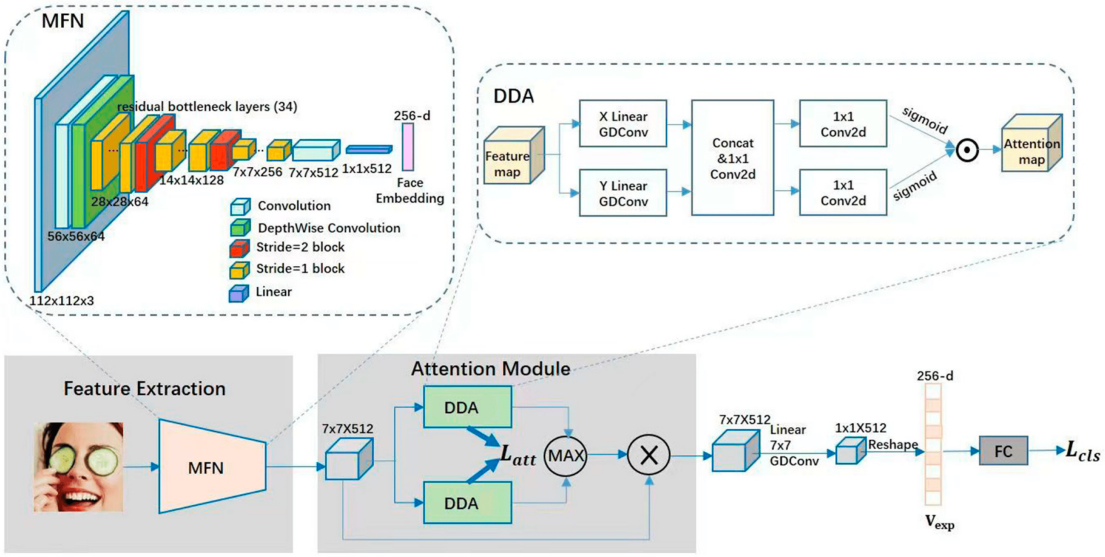


Figure 2.9 Architecture of the novel DDAMFN network that comprises 2 parts – MFN as backbone and DDAN. [80].

Chapter 3
Methods

This chapter describes novel datasets, introduces the design of the system, defines the architecture of networks and proposes benchmarks and experiments.

3.1 Proposed framework

As mentioned in section 1.1, the entire system will consist of three components: face detection, image spectrum translation, and facial expression recognition. The system will be developed using two alternative approaches, which are described below.

Approach 1: First, the input video in NIR will be split into frames and subsequently processed by the face detection system. Then those frames will be translated to the visible spectrum images. Finally, the images will be evaluated with an existing facial expression recognition system from VIS [40].

Approach 2: As the first approach consists of 3 parts, it will presumably have a longer inference time. Therefore, the creation of a system that can evaluate facial expressions from NIR images will enhance the inference time.

There are numerous justifications for implementing a system with *Approach 1* in the first place. First, the translation from $NIR \rightarrow VIS$ will be created as a side product when developing *Approach 2* (the CycleGAN architecture produces both direction translators), thus it is worth testing on.

Second, it can work as a backup if *Approach 2* doesn't work as expected, namely the facial expression recognition from the NIR image. Also, the testing of this approach assumes a very good translation from $VIS \rightarrow NIR$ (even though the translation is tested with the benchmark, it is not easily assessed with metrics) and that is something that *Approach 1* does not need to tackle with.

Both approaches are depicted in a figure 3.1, where subfigures *1)* and *2)* demonstrate *Approach 1* and *Approach 2* respectively.

3.2 Custom datasets and expression annotations

A new database of NIR images has been created for validation and training of models - named **CustomDB**. Also, images from BUAA db with non-neutral expressions were annotated and the OuluCasia images had been automatically assigned valence/arousal labels. For the image spectrum translation was created a small modest database CustomMorphSet.

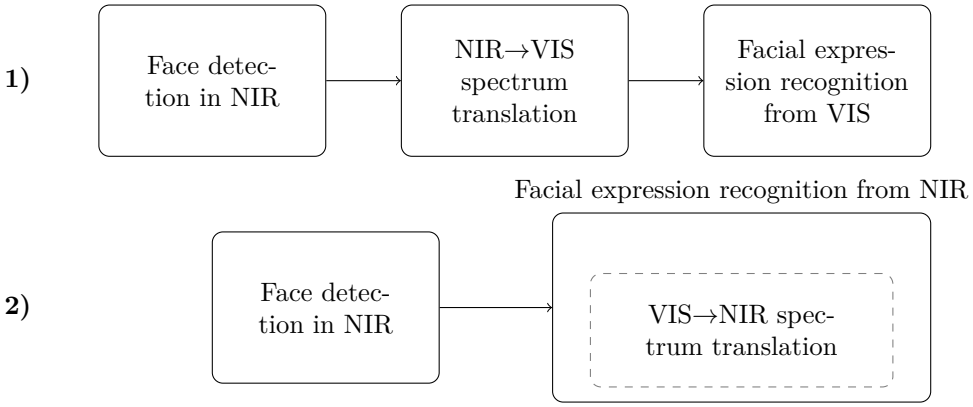


Figure 3.1 The diagram describes two approaches for this task. Bold rectangles represent use of model during inference, whereas the dashed rectangle represents the model, which was necessary for training its parent model, but not used in inference.

3.2.1 CustomDB - Protocol

All subjects were instructed to do the following expressions:

- neutral
- happiness
- sadness
- surprise
- fear
- anger
- 2× disgust
- 2× contempt
- 2 random expressions except those above (such as tiredness, envy, calmness, love, boredom, etc.)

For each expression, multiple images were acquired and from those, one image (or two images) from each expression was chosen. For disgust and contempt, 2 images were extracted (due to the lack of those categories in other databases). For all subjects, overall 12 images were chosen. Expressions were acquired from 19 subjects which makes 228 images, however, some images had insufficient expression so the total number is 193. All subjects were of Caucasian race in the age range of 21-60 and their gender was 9 women, 10 men. Examples of images are depicted in figure 3.2.

Images were acquired from NIR camera TP-LINK Tapo C500 with 1080p resolution, although, a few images were in a lower resolution.

Images were subsequently processed by either the CenterFace from the *Original* work or RetinaFace face detector (including face alignment) mediated by deepface framework to 224x224 images. Then annotated categorically and spatially (Circumplex Model of Affect) using annotator from the original work. The annotations and it's distribution are depicted in figures 3.3.

3.2.2 BUAA annotations

Furthermore, by utilizing the annotator from the original work and the BUAA database, which consists of four facial expressions per subject, more images were annotated. To be precise, 634 additional images were annotated, and their distribution is illustrated in Figure 3.4.

3.2.3 OuluCasia annotations

Although this database already has the categorical labels of expression, it does not have the spatial valence/arousal labels. Also, those images do not have the neutral and contempt expressions. Since there is a large amount of those images, spatial annotations and neutral labels have been assigned automatically, inspired by [4].

Every set of images per patient's expression has approximately 20 images starting from neutral and gradually developing into the most affected expression. For the assignment of valence/arousal labels, the most suitable formula for the actual transition in those image sets was the following one. The first quarter had been assigned neutral values (0,0), and the rest followed the exponential formula defined in equation 3.1.

$$E(t) = \text{anchor1} + (1 - (1 - t)^e \cdot (\text{anchor2} - \text{anchor1}) \tag{3.1}$$

,

where e is the base of the natural logarithm, and $t \in [0, 1]$. When $t = 0$, we get *anchor1*, and when $t = 1$, we get *anchor2*. This formula will give an exponential transition from *anchor1* to *anchor2*, in this case between the neutral label and the most affected image in a particular image set.

As for the categorical labels, the first quarter had been assigned the neutral label and the rest had assigned expression of that particular set. The expressions have been anchored to valence/arousal values captured in the table 3.1 according to AffectNet.

Emotion	Valence	Arousal
Neutral	0.00	0.00
Anger	-0.44	0.8
Disgust	-0.64	0.50
Fear	-0.1	0.8
Happiness	0.92	0.15
Sadness	-0.84	-0.40
Surprise	0.3	0.85

Table 3.1 Valence-Arousal values for the affected expressions.

Other strategies were tested, but this happened to be the most suitable since the graduation is not completely linear which most of the studies assumed. Also, the graduation of affection varied throughout every set of images, so those assigned labels might not correspond to reality for every set of images.

3.2.4 CustomMorphSet

This database was created from the CustomDB and OuluCasia datasets inspired from [4]. It is a dataset of morphed faces of the NIR images. This aims to increase the number of NIR images, particularly of the caucasian race, where it can be utilized in Image spectrum translation.

For morphing images, MorphSet software from GitHub was employed [49], which utilizes the Dlib [29] library for Facial landmarks detection, Triangular Delaunay segmentation, Affine transformation and Alfa blending. This way, 184 new images were created, morphing between the same person and different people. All images were filtered so they were realistic and suitable for the task. An example morphed image is depicted in figure 3.5; face A has a jealous expression whereas face B has a disgusted expression. However, those new images were not annotated.

Altogether, 827 images of varying facial expressions were annotated. Nevertheless, classes and valence/arousal labels are not balanced, especially in images from BUAA db. Another 184 unannotated images were created as CustomMorphSet.

3.3 Design

This section describes how the sub-systems work separately.

3.3.1 Face detection

For face detection has been used pretrained detectors - CenterFace and RetinaFace. CenterFace model was used from the original work and RetinaFace was mediated by the framework DeepFace [60], which also provides face alignment. Testing of face detection was done with the designed face detection benchmark discussed in this 3.5.1.1 subsection. Other models have been experimented with, such as Viola-Jones's algorithm or MTCNN, and are easily implementable in the provided solution. It is worth noting, that the RetinaFace detector detects faces with vertical black side-stripes. That is because the face is detected as a rectangle and when the output needs to be of square dimensions (as an input to the network). On the other hand, CenterFace detects faces (almost) of square dimensions. The difference can be seen in figure 4.2 so there is no confusion about the side-stripes.

3.3.2 Image spectrum translation

In the field of spectral image translation, several network models were evaluated, including DenseUNetGAN, FFE-CycleGAN, and the original CycleGAN, as suggested by recent studies. These models were mostly exclusively trained and assessed on a single dataset. One of the papers also tested models on images of patients, that were in training set, which is not good practice. The dataset's lack of variety and size led to a tendency for the models to overfit (results shown later), which means they performed well on the training data but were unable to generalize to new, unseen data effectively. This was proven by several preliminary experiments, where trained models lacked generalization.

To address this, a combination of datasets was gathered, including Oulu-CASIA, CASIA 2.0, BUAA, AffectNet, and a custom-created datasets CustomDB and CustomMorphSet.

Upon review, it was determined that CycleGAN was the most suitable model for this task. Not only that the CycleGAN produce both directional translators but also because that architecture does not require paired pixel-to-pixel images for translation, unlike the DenseUNet and FFE-CycleGAN models, making it more versatile for different data types of images. Even though the OuluCasia, CASIA and BUAA are paired NIR-VIS datasets, they performed poorly. The first two are not completely pixel-to-pixel pairs and the BUAA database has VIS light images enlightened by green light[1]. For the optimal performance of CycleGAN, the following was tested

[1] Suitable for face recognition.

where the text highlighted in bold had the biggest impact and the text in parenthesis was the best setting used in training:

- **double optimization** in both generators per iteration,
- hyperparameter λ – multiplier of cycle-consistency loss [$\lambda = 10$],
- weight initialization [Kaiming initialization],
- optimizers and their hyperparameters [Adam, beta1 = 0.5],
- **learning rates** and batch sizes [learning rate = 0.0002, batch size = 1],
- generators and discriminators architecture [for both **ResNet9**],
- preprocessing of images[2].

Multiple subsets of all gathered images were experimented with and are described in subsection 3.5.2.1

Applying the model on VIS images generating the NIR images is utilized in *Approach 2* as a translation of FER databases - in this case, the AffectNet database. This will later be used as data for the development of the FER model. On the other hand, as the CycleGAN trains both translation models, applying the model on NIR generating VIS images can be utilized in *Approach 1* – first, the face detected from the NIR image, then the translation to the VIS image and finally FER on the output image. Both approaches are described in section 3.1.

3.3.3 Facial expression recognition

The method for Facial expression analysis was built in the *Original* work – using the MobileNet network, described in theoretical section 2.4. This model predicts both categorical labels of categorical expressions and spatial labels for valence arousal labels. The MobileNet model weights (in 3.5.2.2) were pre-trained on two separate NIR datasets and their variations, resulting in models trained on different data.

The first dataset, AffectNet, was transformed into the NIR spectrum (subsequently referred to as AffectNetNIR) using the model from Experiment1.2, which is further described in Section 3.5.2. Additionally, the model from Experiment1.1 was utilized for a different translation to create another variant of AffectNetNIR (to be described later); however, the version from Experiment1.2 is prioritized. The AffectNet already had provided the train/test/validation splits and they were slightly adjusted to 75%, 5% and 20% [3].

The second dataset is a combination of OuluCasia (it's NIR images only), CustomDB and BUAA datasets. To keep this dataset resistant towards overfitting, only images that differed were chosen. That means, only 3 images per set per patient were randomly selected in OuluCasia. From the BUAA images, only neutral and expression types of images were chosen and CustomDB remained whole. This *Combined dataset*, as is later referred to, was also divided into train, validation and test splits (72%, 13%, 15%), where the same people were not in another split. The size of the *Combined dataset* is depicted in table 3.2. Also, the distribution of expression is captured in table 3.3.

Both models trained on separate data were tested on each other – the *Combined dataset* can be tested on AffectNetNIR and vice versa. Hence, the balance of the train, validation and test split in the combined dataset can have a bigger train split.

[2]Tested were images with square size and images with rectangle size, where black stripes by sides filled the image to be of squared dimensions. The idea behind this adjustment is to force the network to focus more on the facial area and not the background.

[3]Splits might seem imbalanced, however, the AffectNet database is huge and this proved to be a suitable proportion.

database	# train	# test	# validation
BUAA	522	93	111
CustomDB	140	20	33
OuluCasia	1080	144	216
Total	1742	257	360

Table 3.2 Splits size in *combined dataset* per database.

	neutral	happiness	sadness	surprise	fear	anger	disgust	contempt
count	680	390	295	240	260	205	245	50

Table 3.3 Expression distribution in *combined dataset – BUAA, OuluCasia and CustomDB*. The numbers are close approximations.

As the *Original* work introduced, the training process involves using class weights to handle a class imbalance in the dataset. They assign a higher weight to under-represented classes, such as contempt, and a lower weight to over-represented classes, such as happiness. This ensures that all classes contribute equally to the training process, improving the model's normalized metrics, as suggested in the *Original* work. The class weights are computed as an inversion of class size proportion and are subsequently used in optimization as a weighted loss.

As the valence/arousal annotations are imbalanced as well, the *Original* work tested options to employ sample weights as for the categorical labels. However, as was suggested, applying weights is not as straightforward. The first method was to assign weights per region according to class prevalence. This did not improve the model, however, "predictions are somewhat more consistent" as was claimed in the Original work so it was incorporated into this work as well.

Also, as was tested in the Original work, simultaneous prediction (predicting both categorical and spatial labels in one model) does not significantly affect the performance, only the regressor performs marginally worse. Thus, only the simultaneous predictor was used in this application so all the models in this work predict both categorical and spatial labels from images of input size 224×224.

The code for the training and testing is an adjustment of the Original work. There was added (not only) the Optuna[4] study to tune the hyperparameters and the optimal results were the following:

- batch size - 38,
- freezing all the layers except the last 16,
- learning rate - $8e-5$ with exponential decay of rate 0.85.

The parameters mentioned above were used for every MobileNet models trained on AffectNetNIR and the *Combined dataset*.

During the model training, data augmentation techniques were employed. The brightness of images was adjusted within a range of 0.9 to 1.1. A zoom range of 0.8 to 1 was used, and images were rotated within ± 10 degrees. Horizontal flipping was also applied. These techniques, although not critical due to sufficient data, helped improve the model's performance.

Additionally, because (most of the) models were trained images detected by CenterFace – which detects without side-stripes (difference with and without side-stripes observed in 4.2), further augmentation was employed. For the model's independence on the face detector (which factually means independence on the presence of side-stripes in images) was added augmentation

[4] Hyperparameter optimization framework.

which randomly add side-stripes to the image so that are represented equally both CenterFace and RetinaFace detection style images. This was added as the preliminary experiments proved that when trained on only one type, the performance on images of the other type was worse than the first one.

3.3.3.1 DDAMFN

The second architecture that was experimented with was the DDAMFN network described earlier in the theoretical section. This network provides the state-of-the-art solution, however, for the cost of a significantly longer training process slowed down by the employment of an attention mechanism. For the development was utilized repository from GitHub [62] which was then adjusted to custom needs. Whereas the DDAMFN predicts only categorical expressions, a regression output was incorporated in the same manner as in the MobileNet, which means adding to the last non-output layer a regressor layer of 2 neurons with a sigmoid function for valence and arousal. The augmentation of data was the same as for the MobileNet, the class weights were added as well and only the weighting of valence arousal labels was not implemented.

Hyperparameters were the following (inspired by default values):

- batch size - 128,
- learning rate - 0.0001 with linear decay after the first epoch,
- optimizer - Adam,
- epochs - 40.

Also, the loss function and evaluation metrics remained the same as in the MobileNet – this is discussed in the following lines.

3.3.3.2 Loss function and evaluation metrics

Since the model predicts both categorical and spatial labels, the loss function needs to capture both losses of those labels. For categorical labels, the categorical cross-entropy loss (described in subsection 2.1.3.3) was used and for the regression of spatial labels, a cross-entropy loss is used as well, which was introduced in [17] for predicting valence/arousal labels. Since the cross-entropy expects the input to be between 0 and 1, the values of valence arousal are predicted by sigmoid, which meets this condition (and later for inference is recalculated back to a range of the Circumplex model of Affect. Both losses were then combined into one by averaging.

For the validation and testing purposes were tracked multiple evaluation metrics – *MSE*, *MAE* and *RMSE* for the validation step. For testing the classifier, first top 3 *accuracies* and *F1-score*. For the regression, the metrics commonly tracked for *FER* – *RMSE*, *SAGR* and *CCC* metrics are computed for both valence and arousal.

Although accuracy is a very common metric in classification tasks, it might not be perfectly suitable for imbalanced datasets, which happens to be this case – some classes are represented 20× less, such as contempt expression. For this case was used *200-fold stratified down-sampling* – a method that randomly picks n samples from each category (*strata*) and subsequently calculates test metrics on those samples, finally this process is repeated 200 times. The final result is the mean of computed metrics – accuracy and F-1 score.

For closer evaluation of spatial predictions is employed normalized RMSE. This refers to the value, that is a mean of non-empty regions in the circumplex model of affect. There are 71×71 regions in that valence/arousal space.

Note, that this is designed in the *Original* work.

3.4 Realisation

The code is available on *GitHub* in the following repository – https://github.com/kalabto2/Facial-expression-analysis-from-NIR-image, which includes robust Inference module.

3.4.1 Technologies

For the development was used Jupyter Notebook on Python3 and its scientific packages – Numpy, Pandas, MatplotLib, Open and such. Regarding machine learning packages, Keras [6] API on the TensorFlow platform for developing MobileNet and Pytorch [46] for others. Also, hyperparameter optimization framework *Optuna* [1] was employed. It significantly decreased the time spent on tuning.

As for the hardware, models have been trained, validated and tested on *Google Colab*'s NVIDIA T4 GPU and *CTU*'s NVIDIA GeForce RTX 2060 SUPER GPU.

3.4.2 Inference

For easy manipulation, a Python module called *FaceInference* was created with more than 1000 lines of code. It is easily importable and can be quickly defined and run when the module is exported to *pip* – the code snippet 3.1 demonstrates the installation process. The idea behind the inference module is that it connects face detection, spectrum translation and FER in a pipeline with wide options for loading and saving data, preprocessing and displaying. Also, models for each part can be selected according to needs.

```
# Create virtual environment and switch into it (optional)
virtualenv venv
source venv/bin/activate

# Change directory from the root of the repository to the 'skeleton/'
cd skeleton

# Install Python Module locally
pip install .
```

Code Snippet 3.1 Installation process of Python module locally.

For displaying the expression was created dashboard capturing both categorical and spatial labels. The design is a modification of the original work and is depicted in figure 3.6. Also, for processing video records of people, a feature was created that displays the FER results right into the original video, as is depicted in the figure 3.7. It displays the top 3 expression predictions and valence-arousal value.

More extensive documentation with the usage examples is in the repository in branch baseline and file inference.ipynb.

3.5 Experiments and Benchmarks

This section introduces the conducted experiments and testing benchmarks for face detection, image spectrum translation and FER. It introduces the methodology and design of experiments and benchmarks. The results are described in the *Results* chapter.

3.5.1 Benchmarks

Two benchmarks were designed – for face detection and image spectrum translation. Those should effectively evaluate how models detect faces and measure the concordance between NIR and VIS FER models. The results of the benchmarks are in the chapter Results.

3.5.1.1 Benchmark for Face Detection

The benchmark was conducted on BUAA database[5] that has pixel-to-pixel paired NIR and grayscale VIS images containing 1950 images. Two face detectors were tested – RetinaFace and CenterFace; mean IOU and mean inference time[6] were tracked for assessing the performance.

Although the detection of images is done with grayscale VIS and not RGB VIS, the benchmark should have some predictive value, since these face detection models were trained on a large amount of data (database *WIDER FACE* [74]) scraped from the internet containing also grayscale VIS images.

3.5.1.2 Benchmark for Image Spectrum Translation

To address the correctness of $VIS \rightarrow NIR$ spectrum translation utilized in **Approach 1** (3.1), the benchmark for testing this translation was designed where the paired OuluCasia dataset was employed as a testing set. The *Original* model on AffectNet evaluated VIS images and then the new NIR model evaluated its NIR counterparts. Multiple NIR models pretrained on the Original VIS model were tested and compared with the VIS model. Subsequently, the following concordance metrics for the classifier were calculated: total concordance percentage, concordance percentage per category and *Cohen's Kappa* (described later in this section). For the regression of valence/arousal labels, the CCC coefficient and average distance between the label values (for a better idea of difference) were calculated.

Furthermore, the metrics were also calculated on several subsets of OuluCasia (the whole dataset, the upper half of each set and the upper quarter of each set) to address the ambiguous images at the beginning of the sequence in each set for every emotion per patient.

Additionally, Chen et al. [4] evaluate the translation quality by comparing the results between the model trained on VIS AffectNet and NIR AffectNet. In their case, the model was translating well according to the similar accuracy values of those 2 models. As the *Original* work already trained the MobileNet model on the VIS, it is also easily applicable to this case, which is further discussed later.

The benchmark for translation $NIR \rightarrow VIS$ (which is employed in *approach 2*) was also created – it measures the concordance between the original model on VIS data and the original model with NIR data translated to the VIS. Nonetheless, this one needs to be taken less seriously, because the spectrum translation model was also trained on the part of NIR images from the OuluCasia dataset [7].

Definitions of the evaluation metrics:

Kohen's Kappa: Cohen's Kappa (κ) is a statistic that measures inter-rater reliability for categorical items. It is generally thought to be a more robust measure than simple percentage agreement calculation, as κ takes into account the possibility of the agreement occurring by chance. The formula for Cohen's Kappa is (in formula 3.2):

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{3.2}$$

[5] Only well-illuminated images were used.

[6] Testing was performed on Intel Core i5-6300HQ @ 2.30GHz

[7] It doesn't make much sense to use the OuluCasia images from a small test set for the spectrum translation model.

,

where κ is Cohen's Kappa, p_o is the relative observed agreement among raters (also known as observed proportionate agreement) and p_e is the hypothetical probability of chance agreement.

The value of κ lies between -1 and 1 where value of 1 implies perfect agreement, and values less than 1 imply less than perfect agreement. A value of 0 implies that agreement is no better than chance and negative values imply that agreement is worse than chance. If the value is > 0.8, it is almost perfect, > 0.4 is a substantial/moderate classifier and < 0.2 is a fair/poor classifier.

Concordance Correlation Coefficient: The Concordance Correlation Coefficient (CCC) is a statistic used to measure the agreement between two variables. It measures both precision and accuracy, making it a more comprehensive metric for agreement. The formula for CCC is (in formula 3.3):

$$\rho_c = \frac{2\rho\sigma_x\sigma_y}{\sigma_x^2 + \sigma_y^2 + (\mu_x - \mu_y)^2} \tag{3.3}$$

,

where ρ_c is the Concordance Correlation Coefficient, ρ is the Pearson Correlation Coefficient, σ_x and σ_y are the standard deviations of x and y, μ_x and μ_y are the means of x and y.

The value of ρ_c lies between -1 and 1 where a value of 1 implies perfect agreement, and values less than 1 imply less than perfect agreement. A value of 0 implies that agreement is no better than chance and a negative value implies that agreement is worse than chance.

3.5.2 Experiments

This subsection briefly summarizes the most important conducted experiments for the FER and Image spectrum translation. The results of those experiments are described later in chapter *Results*.

3.5.2.1 Image spectrum translation

After extensive preliminary experiments, the most notable experiments were those on the following lines.

Experiment1.0: CycleGAN on *Data1.0* for 100 epochs + 100 epochs of decay with double optimization of generator per iteration. This experiment on the OuluCasia dataset is an established baseline in the $NIR \leftrightarrow VIS$ translation field of research.

Experiment1.1: CycleGAN on *Data1.1* for 100 epochs + 100 epochs of decay with double optimization of generator per iteration and ResNet6 architecture for generators. This experiment builds upon the established experiment (in this case it is *Experiment1.0*) with training on additional datasets.

Experiment1.2: CycleGAN on *Data1.2* with cycle-consistency multiplicators $\lambda_{NIR} = 15$ and $\lambda_{VIS} = 10$, ResNet9 architecture for generators with double optimization for 10 epochs + 15 epochs of decay. It is an adjustment of *Experiment1.1* – adjusting the training data and lambda values.

The hyperparameters used are as in 3.3.2 if not specified else, and the *Data0*, *Data1.1* and *Data1.2* are described on the following lines.

database	#NIR	#VIS	#NIR mongo	#NIR cauca	#VIS mongo	#VIS cauca
AffectNet	0	2054	0	0	0	2054
BUAA	1739	0	1739	0	0	0
CASIA 2.0	662	664	662	0	664	0
CustomDB	151	0	0	151	0	0
OuluCasia	2160	2164	840	1320	841	1323
MorphSet	170	0	0	170	0	0
Total	**4882**	**4882**	**3241**	**1471**	**1675**	**3377**

Table 3.4 Training set distribution by database and race for spectrum translation.

Data1.0: This data is a random 5-image-sized subset of each set of emotions per patient from the OuluCasia dataset. The splits were train set (85%) and test set (15%)and faces were detected on both CenterFace (without side stripes) and RetinaFace (with side stripes) detectors.

Data1.1: This data includes in the NIR spectrum the OuluCasia, BUAA and CASIA 2.0 databases and the images in a VIS spectrum include only the OuluCasia and CASIA 2.0 databases – using then all the databases that were obtained. Again, it was used only a subset of databases and split to train and test set in the same ratio as in **Data1.0**. The faces were detected with CenterFace, thus without side-stripes.

Data2: In the NIR spectrum, this data had the same data as **Data1.1** and added CustomDB and CustomMorphSet images. In the VIS spectrum, however, it was used only *AffectNet* images and a minority part of *CASIA 2.0*. The use of AffectNet aims to improve the generation of caucasian faces, enhance generalization and prevent overfitting. Faces were detected with both CenterFace and RetinaFace, hence the model is trained on images both with and without side-stripes (as discussed in Face detection section 3.3.1). This data comprised all gathered images and was split into the train (90 %) and test set (10%). The split is rather imbalanced, however, the translation from VIS to NIR can be partly evaluated on AffectNet and from NIR to VIS on left-out CASIA 2.0 images[8]. The number of samples per dataset and race distribution is depicted in table 3.4.

Proposed experiments focus on the training data because all of the unpaired models by other researchers do not merge datasets (described in the research chapter 1.3) and use only one dataset. Therefore this might be a way how to improve the translation.

3.5.2.2 Facial Expression Recognition

As mentioned earlier, the models of MobileNet were trained on two separate datasets (and it's variations). The hyperparameters for those models are described earlier in 3.3.3 and the main experiments after extensive preliminary studies are the following:

Experiment2.1 MobileNet trained on *AffectNetNIR* (pretrained on the model from *Experiment1.2*) for 15 epochs. When employing the MobileNet, performance can be also compared to the model in an existing solution.

Experiment2.1.0 This is the same experiment as the *Experiment2.1*, but with the AffectNet-NIR translated by the model from *Experiment1.1*. Also, this model was trained only on images without side-stripes (detected by CenterFace), thus is less effective on images with side-stripes (detected by RetinaFace).

[8]Those images were all of the Mongolian race and would distort the translation if used in the training dataset.

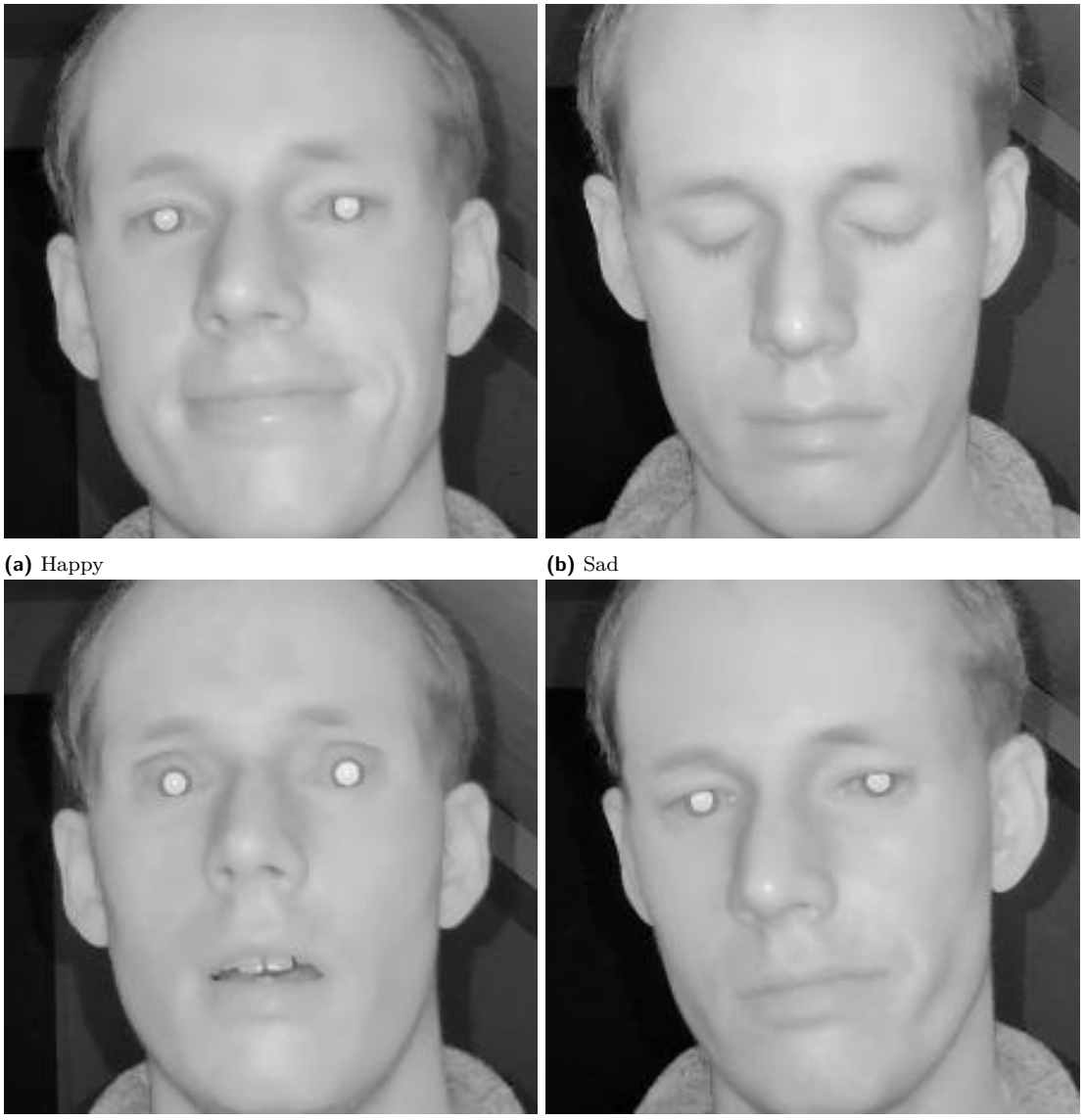
Experiment2.2 MobileNet trained on *Combined dataset* (BUAA, OuluCasia and CustomDB datasets) for 15 epochs (stopped earlier). Although the AffectNetNIR dataset is very large and with a wide variety, the images are generated from VIS spectra, thus this experiment aims to create a supporting model trained on true NIR images used mainly to evaluate and ensure validity of the first experiment.

Experiment2.3 This is the same experiment as the *Experiment2.2* but without the OuluCasia data. That makes the training set only from custom annotated images. Mind, that even though the amount of data without OuluCasia is very small (≈ 600), it is worth trying since the model is pretrained on the *Original* model that was trained on a very extensive dataset. Furthermore, the results can be compared to *Experiment2.2*, thus estimating the quality of assigned annotations of the OuluCasia dataset.

Experiment2.4 DDAMFN architecture trained on the AffectNetNIR translated by the *Expreiment1.2*. Training on another AffectNetNIR version is unfeasible because the training lasts several days, almost a week. Additionally, employing the small *Combined dataset* is not suitable for this network – it would be easily overfitting.

Experiment2.5 This experiment is only testing of **Approach 1** where the NIR image is first transferred to the VIS spectra employing the **Experiment1.2** and subsequently classified with the *Original* model.

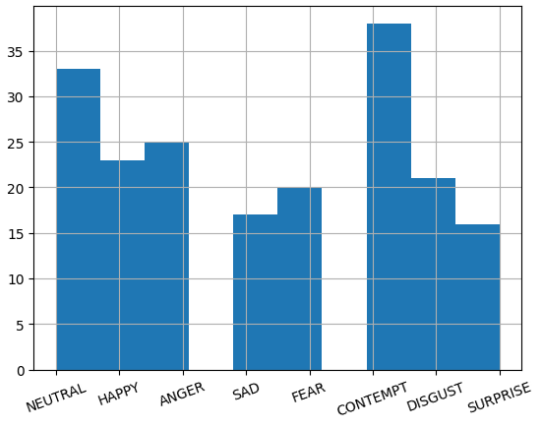
Each Mobilenet model used the following weights in the original code repository for pretraining: *src/model/mobilenetv1_05_rgb_simultaneous_sigmoid_ClassWeights_CHECKPOINT.hdf5*.



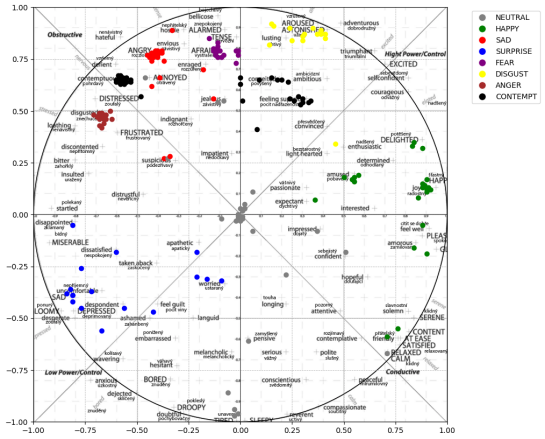
(a) Happy **(b)** Sad

(c) Fear **(d)** Contempt

Figure 3.2 Example images from CustomDB.

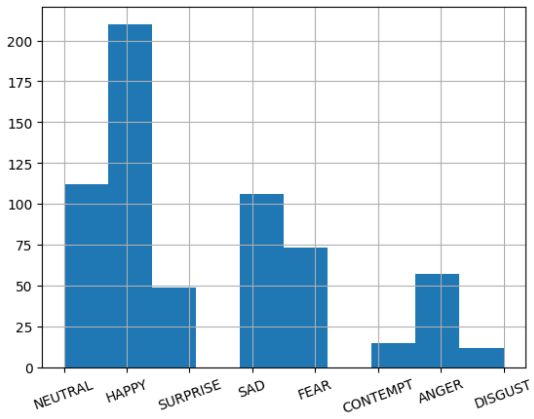


(a) Categorical distribution of expressions in Custom database

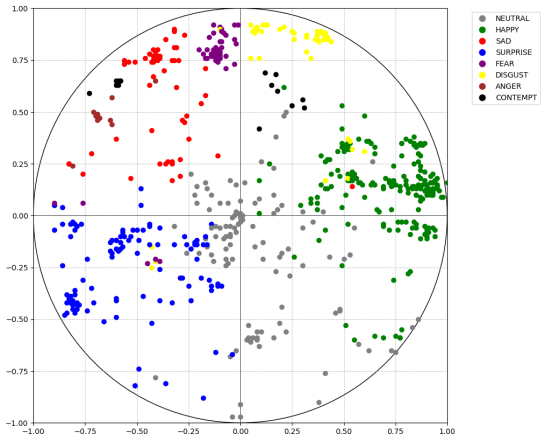


(b) Spatial valence/arousal labels in Custom database

Figure 3.3 Image captions distribution of annotations in CustomDB.



(a) Categorical distribution of expressions in BUAA database



(b) Spatial valence/arousal labels in BUAA database

Figure 3.4 Image captions distribution of annotations in BUAA expression images.

(a) Face A **(b)** Face B **(c)** New morphed face

Figure 3.5 Image captions newly created artificial face morphed from 2 distinct faces.

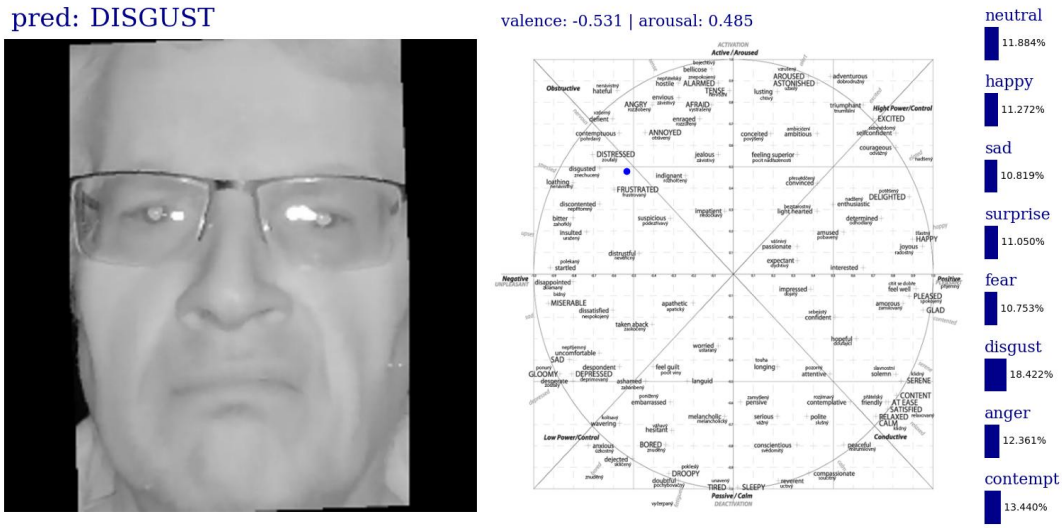


Figure 3.6 Image captures an example of the created dashboard of affect for a single image.

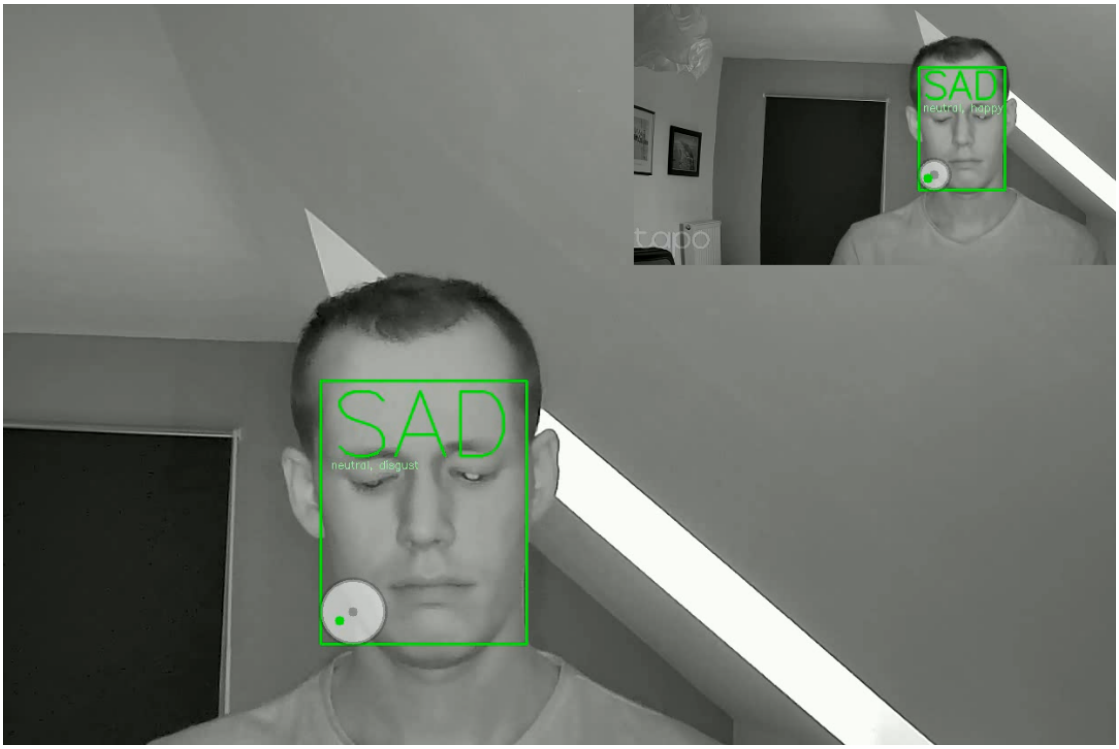


Figure 3.7 Image demonstrates one frame of video processing feature in the Inference module. Frame captures the face detections and results of FER for each detected face. The "double frame" is not the result of inference, but is a visual of the input video. It serves to demonstrate that inference can handle multiple faces in different proximity.

Chapter 4
Results

The chapter reveals the results and describes the observations of every experiment/benchmark and mildly elaborates on them. Interpretation and discussion of the results are then mostly left for the *Discussion* chapter.

4.1 Experiments

This section describes the experiment results of both image spectrum translation and FER described in subsection 3.5.2.

4.1.1 Image spectrum translation

In figure 4.1 are depicted results of the **Experiment1.0** – an established baseline experiment in the field of $NIR \leftrightarrow VIS$ translation. In the upper row are source images and below are translated images. Overall, the model generates good-looking images (column *good*). However, the images are mostly overfitting and the model is generating images corresponding to its opposite spectrum counterpart, which are not completely from the same viewpoint. Sometimes, the model even generates image of another patient. This can be seen in the figure in columns *overfit*. Occasionally, the model generates bad-quality images, that can be visible in *bad* column. These results resemble those in other studies which are overfitting as well. Testing on another dataset proved overfitting, so this baseline model is not usable.

Figure 4.2 [1] depicts the results of the rest of the experiments. The first three rows portray translation from $VIS \rightarrow NIR$ translation, which is used in **Approach 2** (described in section 3.1). **Experiment1.1** has more contrasting images compared to images from **Experiment1.2**. Generally, the first experiment resembles more standard grayscale images than the second one, resembling NIR images from the training data. The difference of contrast can be seen in the eyes of the face in images *(b)* and *(c)*. Also, the first experiment has (occasionally) artefacts, caused by so-called mode collapse, as in image *(j)* and has an insignificant grid-like artefact from a closer look, which the second experiment does not have. The resolution of those images in the second experiment is not the same as the original image, however, it is a very satisfactory result. Finally, it is worth noting that the original images vary by the sensor in the camera as can be seen in the figure 4.2 – subfigures *(m)* from the CustomDB vs *(q)* from the BUAA.

Regarding the $NIR \rightarrow VIS$ translation, the results of experiments are in the last 3 rows of the figure 4.2. From the provided images is clear, that the **Experiment1.1** has worse results

[1] Note that models for translation to VIS and NIR are not from the same epoch. Also, the side-stripes are just the result of using the RetinaFace detector.

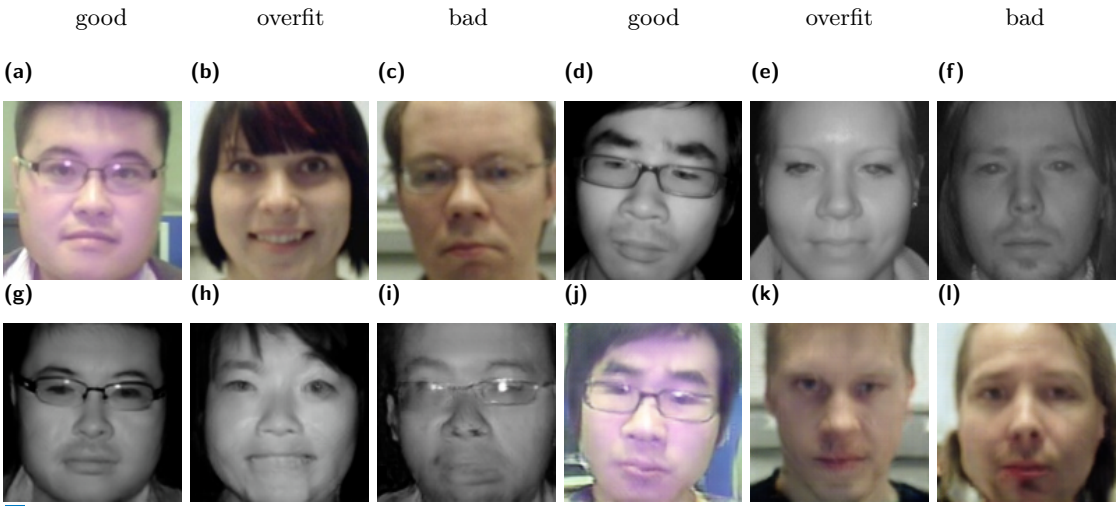


Figure 4.1 Figure depicts results of **Experiment1.0** on OuluCasia dataset. In the first row are source images and translated images in the second row. Columns depict *good*, *bad* and *overfit* examples described in the text above. When looked closer, all images overfit to some extent.

than the second experiment. The first experiment generates images resembling the training data (OuluCasia and CASIA 2.0) which are both lower quality and have low variety (especially the OuluCasia dataset). On the contrary, the second experiment has OuluCasia replaced with the AffectNet database during the training phase providing both higher quality and variety making it a better and more robust solution. Also, in each experiment was observed, that the translation of images with mongolian faces had better results overall, resembling the images of the CASIA 2.0 dataset. That might be due to the bigger variety representation in NIR and VIS mongolian faces and the fact, that they are paired (but not aligned, only the same face), which might speed up learning. Although the images in **Experiment1.2** look appealing, most of the generated images with caucasian faces are less colourful and the mongolian faces are of slightly lower quality than the one depicted in the subfigure *(o)*. However, still satisfiable, has a way better appearance and is more robust than the first experiment. Additional generated VIS images from *Experiment1.2* are in Appendix A in figure A.1 and in figure A.2 and discussed in the text. The generated NIR images are unnecessary to display, as those on the figure are representative and all of the generated images are qualitatively similar.

An additional way how to assess the quality of $VIS \rightarrow NIR$ is through the proposed benchmark that is discussed in section 3.5.1.2 and its result is in its corresponding section in this chapter.

4.1.2 Facial Expression Recognition

The results of the FER experiments are depicted in extensive table 4.1 alongside the comparable studies – DDAMFN (*SOTA* on VIS spectrum), the *Original* work on VIS spectrum and comparable study [4] (FER in NIR spectrum). The tracked metrics are discussed earlier in subsection 3.3.3.2 – categorical metrics, regression metrics for spatial labels and 200-fold stratified down-sampling metrics. The results and conclusions are described in the following lines.

Experiment2.1.0: Regarding the categorical metrics, *Experiment2.1.0* (row *(e)*) has higher accuracy than the *Original* work on AffectNet (row *(b)*) and the F1 score is of slightly lower value. Those metrics under 200-fold stratified down-sampling achieve better results for categorical prediction, thus, training the model itself seems to be at least equally successful

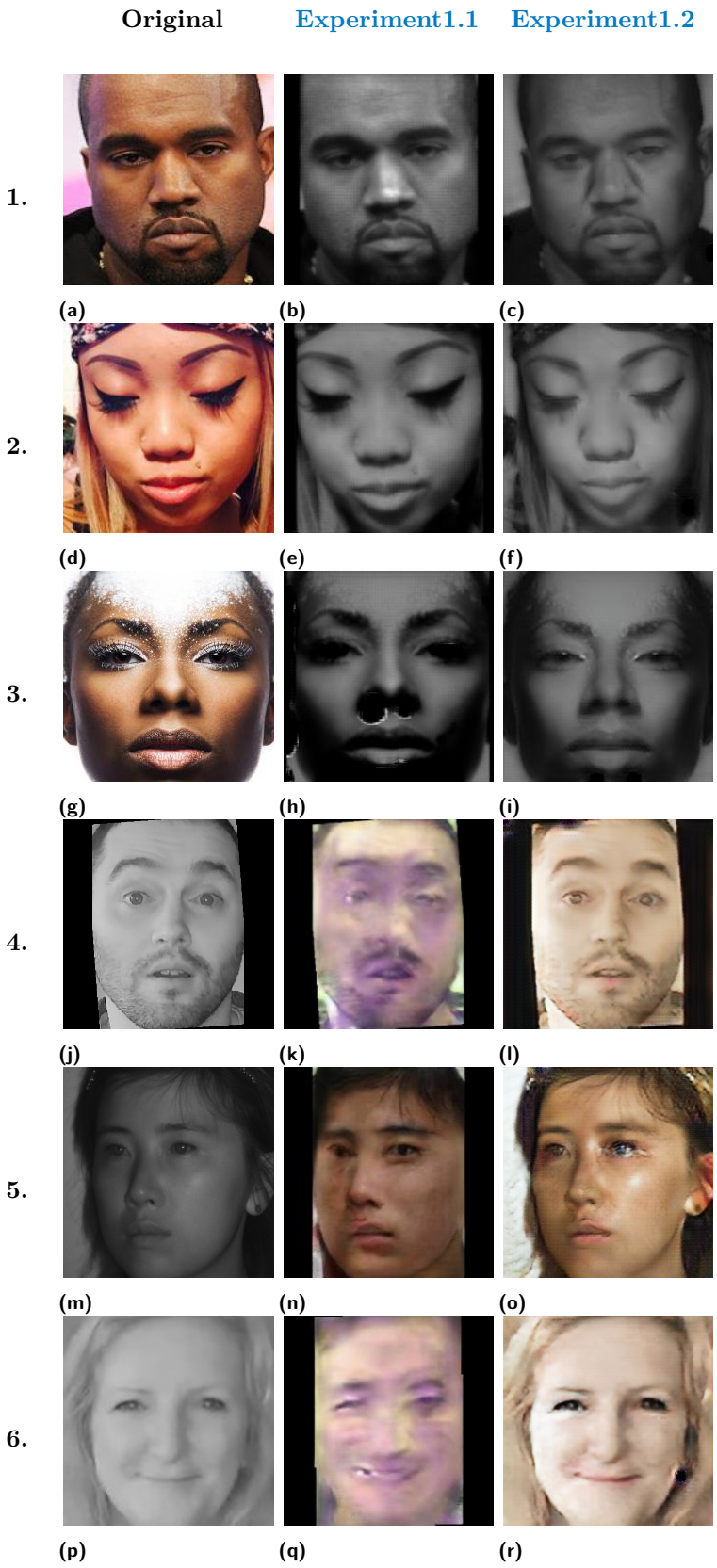


Figure 4.2 Figure captures results of Image spectrum translation between $NIR \leftrightarrow VIS$ (translation from $VIS \rightarrow VIS$ first 3 rows and last 3 rows is reversed translation). The first column captures an original image and the rest of the columns demonstrate experimental results.

as the *Original*. Furthermore, the regression metrics also have comparable values and achieve considerably better results than the study by Chen et al. from 2022 [4] that focused on the same task (FER from NIR spectrum image).

The performance on the *combined* dataset is generally lower than on the AffectNetNIR test set which mostly applies to the categorical labels, the spatial labels are somewhat similar. Analogous to the *Original* work, the custom dataset performs worse, however, in this case, it achieves worse results compared to the Custom160 dataset (test set of the *Original* work) as well.

Figure 4.3 depicts the confusion matrix of the *Experiment2.1.0*. As can be seen at first sight, the grid colouring (determined by total numbers) is very imbalanced, however, that is due to the imbalance of the dataset itself. One can see, that the predictions can be incorrect when the target label is neutral, perhaps because of very close bordering expressions between those two. Also, happiness can be mispredicted for contempt rather easily, where a smile resembles a raised expression of a person. On the bottom row and in the last columns are TPR/FNR[2] and PPV/FDR[3] values respectively. The row with TPR suggests, that the weighting of the samples by category does not oppress the smaller categories in favour of the bigger ones – almost all of the values are above 50%. Overall, all values resemble the matrix from the original work and both the TPR and PPV values for key emotions (neutral, happy, sad and angry) are satisfactory for real-world usage. Furthermore, the heatmap of normalized RMSE values per region is depicted in figure 4.4.

Similarly, the confusion matrix (figure 4.6) and heatmap of normalized RMSE boxes (figure 4.6) were computed for the testing on the Combined dataset as well. The confusion matrix is quite unbalanced with the *contempt* having a low value and the *surprise* high for example. Also, it can be seen the model mispredicts similarly as for the testing on the AffectNetNIR – fear is often predicted as surprise or sadness predicted as neutral and so on. The author hypothesizes that the observed results may be attributed to inadequate acting, given that the performance on spatial labels aligns with that of the AffectNet dataset. This suggests that the network is indeed functional with authentic NIR images. However, to substantiate this claim, further testing on high-quality annotated NIR images is necessary.

Experiment2.1: *The Experiment2.1* had very similar results, where the main difference was in the (normalized) categorical metrics – it had lower normalized accuracy. Nevertheless, the performance is very similar as well, although the previous experiment has a slight edge in the normalized categorical metrics. However, this model The performance per expression category showed in the confusion matrix remained similar to the previous experiment.

Experiment2.2: The experiment trained on the *combined dataset* achieves worse results on the AffectNetNIR compared to previously mentioned experiments. Nevertheless, it achieves better results on *combined dataset* test set (especially on the normalized metrics) with its normalized 49% accuracy and 0.483 of F1 score slightly (2%) leading before the second best.

Experiment2.3: Omitting the OuluCasia dataset from the training data on *Experiment2.2* proved to be valuable because it (mostly) improved performance on AffectNetNIR (both spatial and categorical were improved) compared to *Experiment2.2*. Model on the *combined dataset* improved only the spatial labels – that once again suggests that spatial automatic annotations of OuluCasia are holding down the performance, thus the automatic annotations might not be a suitable solution. On the other hand, the OuluCasia images are useful for the categorical labels – this experiment without the OuluCasia data has worse performance than the previous *Experiment2.2* which contains those images.

[2]True Positive Rate (*TPR*, *sensitivity* or *hit rate*) is a rate of positive labels, that were predicted correctly. False Negative Rate (*FNR* or *miss rate*) is in the same way just for the negative label.

[3]Positive Predictive Value (*PPV* or *precision*) quantifies how often is specific prediction correct. False Discovery Rate (*FDR*) is the opposite of it.

Experiment2.4: Despite the expectations, this model did not achieve superior results – it slightly trailed in normalized accuracy with its 55% just behind the leading 58% of the first experiment and was marginally worse in valence/arousal predictions.

Experiment2.5: This experiment evaluating the success of the *Approach 1* on the AffectNet-NIR had fallen behind Experiments 2.1 and 2.1.0 in the normalized accuracy and normalized RMSE of valence. However, in the arousal predictor, it slightly surpassed all of the experiments including the *Original* model. As for the testing on the *Combined dataset*, spatial labels achieved the best performance tying the Experiment2.1 and the categorical labels trailed after the leading two experiments.

Additionally, the models have approximately the same performance for both types of detectors (the RetinaFace and the CenterFace). The confusion matrices of all of those experiments vary in the TPR per each category, however, most of the categories' hit rates remain above 50% for the AffectNetNIR, for the combined dataset is a little unbalanced. Those matrices and others such as RMSE boxes are stored in project's repository [4]. Other supporting results can be found in *Appendix A* A.

4.2 Benchmarks

This subsection presents results of 2 benchmarks – for face detection and image spectrum translation.

4.2.1 Benchmark for Face Detection

Results are depicted in table 4.2. It is clear, that the RetinaFace detector achieves slightly better detection accuracy for the cost of a significantly slower time, however, the RetinaFace time also includes face alignment. The m-IOU indicates high accuracy, making these face recognition solutions reliable and precise for this application.

4.2.2 Benchmark for Image Spectrum Translation

From the results of the benchmark can be deduced several conclusions. First, the experiments proved that a subset of every emotion set per patient in OuluCasia affects the concordance of the models. This was proven not only with the concordance percentage but with higher Cohen's Kappa values, which do not count class matches "by chance". Thus, the natural deduction is that the less affected images are more ambiguous, the model has problems with neutral expression, the automatic annotations of the OuluCasia dataset are unsuitable, or the images are not at the beginning well paired. This is valuable information – training and testing on the OuluCasia needs to be approached with caution since the annotations might not be correct (and aligned with other datasets' annotations), especially the first images in the sets. The differences between the subsets of OuluCasia were mild – maximal differences between the *all* OuluCasia data and *top-fourth* are 7% and for CCC it is 0.06.

Second, the concordance of categorical predictions is between moderate and substantial according to the study on kappa's statistics [41]. However, in this particular application of FER one might deduce, that those concordance values are more than sufficient when the state-of-the-art prediction is only $\approx$ 65%. Further examination showed, that the concordance percentage distribution of expressions was somewhat similar to the TPR results of FER (section 4.1.2) which can be seen in the table 4.3. The concordance of valence/arousal predictions is surprisingly very

[4] https://github.com/kalabto2/Facial-expression-analysis-from-NIR-image/tree/mobilenet/tests

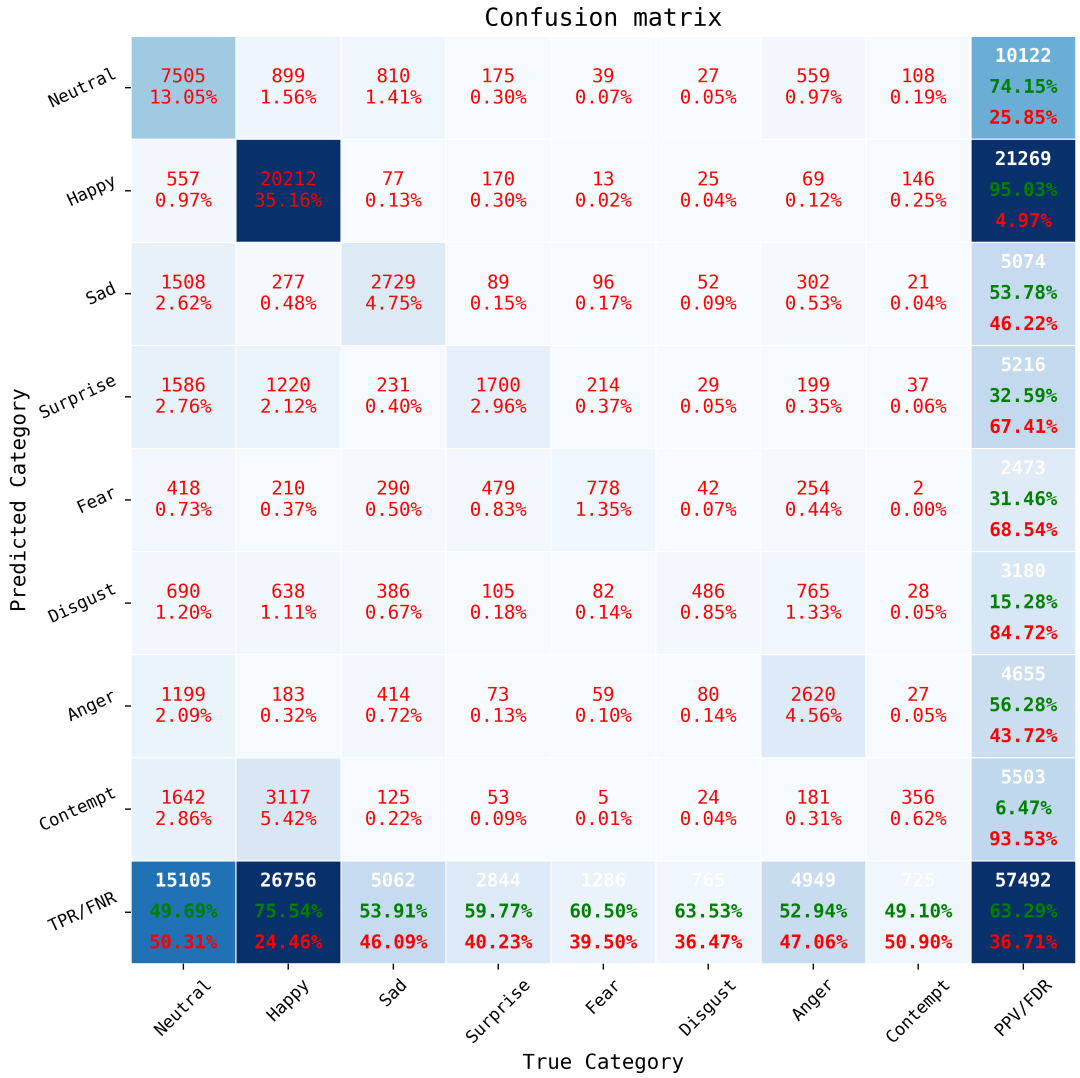


Figure 4.3 Confusion matrix for the categorical predictions of the **Experiment2.1.0** tested on AffectNetNIR. Also, the TPR/FNR and PPV/FDR values are captured. Mind that grid colouring is by the total number of samples in the very imbalanced dataset, thus, that might appear confusing at first sight.

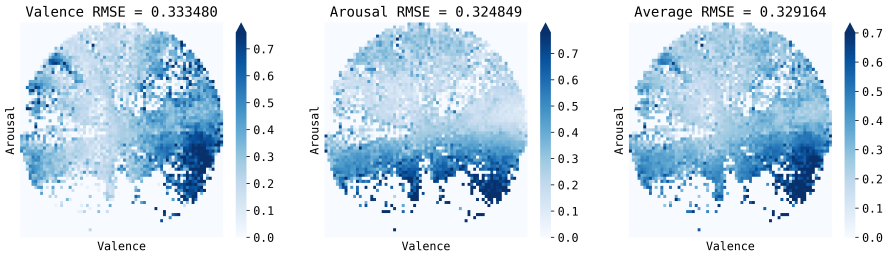


Figure 4.4 Heatmap of normalized RMSE metric per region of the circumplex model of affect for **Experiment2.1.0** tested on AffectNetNIR. Above the figures is also captured the normalized RMSE for valence/arousal/average RMSE.

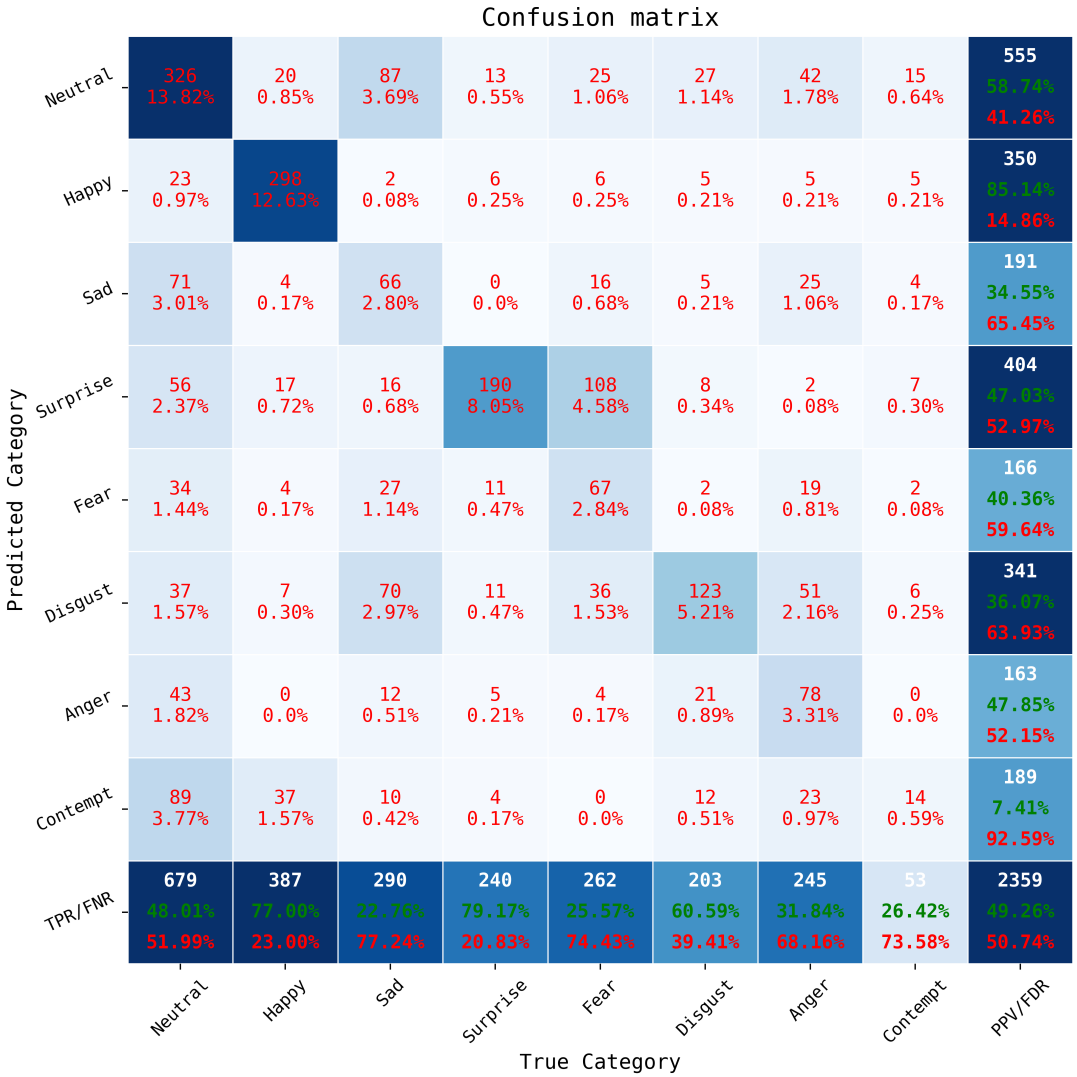


Figure 4.5 Confusion matrix for the categorical predictions of the **Experiment2.1.0** tested on the *Combined dataset*. Also, the TPR/FNR and PPV/FDR values are captured. Mind that grid colouring is by the total number of samples in the very imbalanced dataset, thus, that might appear confusing at first sight.

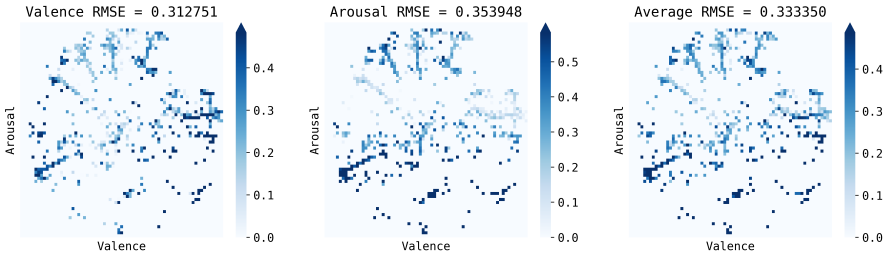


Figure 4.6 Heatmap of normalized RMSE metric per region of the circumplex model of affect for **Experiment2.1.0** tested on the *Combined dataset*. Above the figures is also captured the normalized RMSE for valence/arousal/average RMSE.

id	model [test set]	categorical		spatial (valence/arousal)			normalized		
		top–ACC (top–1/2/3)	F1	RMSE	CCC	SAGR	ACC	F1	RMSE
(a)	DDAMFN [AffectNet][77] - *SOTA*	.647/ -/-	-	-/-	-/-	-/-	-	-	-/-
(b)	MobileNet [AffectNet][40] - *Original*	.620/ -/-	**.470**	.399/**.298**	.644/**.411**	**.755**/**.660**	.580	.580	.342/.319
(c)	MobileNet [Custom160][40] - *Original*	.606/ .775/.868	.549	.339/.399	.665/.477	.743/.650	.554	.537	.273/.321
(d)	EfficientNet-B0 [AffectNetNIR] [4] (2022)	-/ -/-	-	.447/.373	.527/.426	-/-	-	-	-/-
(e)	Experiment2.1.0 [AffectNetNIR]	**.632**/ .812/.902	.465	.377/.303	.693/.403	.752/.653	**.585**	**.584**	.333/.324
(f)	Experiment2.1.0 [combined dataset]	.504/ .711/.835	.442	.312/.308	.658/.361	.631/.604	.473	.449	.312/.353
(g)	Experiment2.1 [AffectNetNIR]	**.632**/ .809/.900	.462	**.376**/.304	**.695**/.404	.753/.651	.573	.573	**.332**/.327
(h)	Experiment2.1 [combined dataset]	.504/ .704/.822	.433	.338/.356	.680/.424	.626/.607	.460	.420	.302/.342
(k)	Experiment2.3 [AffectNetNIR]	.617/ .797/.895	.414	.429/.329	.519/.229	.716/.628	.492	.482	.368/.345
(l)	Experiment2.3 [combined dataset]	.540/ .727/.859	.464	.363/.388	.481/.199	.564/.595	.476	.448	.317/.332
(m)	Experiment2.4 [AffectNetNIR]	.619/ .828/.918	.456	.391/.301	.680/.418	.741/.662	.556	.555	.343/.326
(n)	Experiment2.4 [combined dataset]	.491/ .712/.820	.426	.360/.361	.647/.430	.625/.612	.438	.407	.317/.338
(o)	Experiment2.5 [AffectNetNIR]	.631/ .823/.913	.456	.404/.282	.643/.392	.723/.647	.542	.543	.354/**.317**
(p)	Experiment2.5 [combined dataset]	.475/ .672/.798	.408	.352/.365	.637/.356	.619/.612	.420	.385	.308/.342

Table 4.1 The table depicts the results of experiments and compares them to existing solutions. The first group of models are models on VIS including state-of-the-art (*SOTA*) and the *Original* work. The second group is an experiment on FER from the NIR spectrum – a comparable study to this one, the third group conducted experiments for *Approach 2* and the last group is for experiment representing *Approach 1*. Regarding the metrics, the first group of columns captures metrics for categorical prediction. The second group represents metrics for the prediction of spatial labels. The last column group then 200-fold stratified down-sampling metrics for normalization of predictions. The text in brackets is the test set and grey columns represent less important metrics.

Detector	m-IOU	m-time(s)
RetinaFace	0.95	2.05
CenterFace	0.94	0.11

Table 4.2 Comparison of performance metrics for RetinaFace and CenterFace detectors.

Model	Happy	Sad	Fear	Surprise	Angry	Disgust
Experiment2.1.0	79%	49%	53%	87%	56%	69%
Experiment2.1	77%	51%	55%	89%	54%	67%
Experiment1.2 $NIR \rightarrow VIS$	77%	50%	47%	77%	42%	62%

Table 4.3 The table captures the percentage concordance for each expression of models with the original model. The above concordance values are tested on the top-fourth part of the OuluCasia dataset and faces are detected by the CenterFace. The first 2 models are FER models trained on NIR. The last one is the original FER model that has NIR images translated to the VIS spectrum with the model from *Experiment1.2*.

high with almost perfect CCC values and average distance. The overall results are shown in the table 4.4.

Third, the **Experiment2.1.0** proved to have slightly better results in spatial labels concordance over **Experiment2.1**. That means, probably, that the spectrum translation from **Experiment1.1** is slightly more similar in spatial labels to the original model than **Experiment1.2** because the *Experiment2.1.0* uses data translated by the model from *Experiment1.1*.

Fourth, the concordance is higher on the images detected with CenterFace, which is predictable, since the original MobileNet was trained on the images detected by the CenterFace.

As for testing the $NIR \rightarrow VIS$ translation, the results (tables 4.4 and 4.3) are not as high as the opposite translation, however, still there is at least moderate [41] concordance of categorical predictions, and spatial predictions are comparable as well. Nevertheless, the results might not be necessarily relevant, because the NIR images from OuluCasia were in the training data of that model [5], which were necessary for training the model itself. Therefore, it is important to interpret the results with caution and scepticism. Anyway, the model from *Experiment1.2* had better metrics than the *Experiment1.1*, so it supports the visual findings from 4.2 that it has better results.

Overall, both models translating to NIR from experiments *Experiment1.1* and *Experiment1.2* proved to have at least substantial concordance with the original model. The opposite translation had worse performance but still had at least moderate concordance with the original model. Additional data and findings that are not supported by numbers in this section can be found in *Appendix A* in A.2, A.3.

[5] The test set for evaluating the spectrum translator mode consisted of only a few pairs of images, which made it unsuitable for reliable benchmark testing.

	Categorical metrics		valence/arousal metrics	
Model	**Concordance**	**Cohen's Kappa**	**CCC**	**Avg. dist.**
Experiment2.1.0	65.36	.595	.921/.800	.057/.056
Experiment2.1	65.448	.596	.895/.800	.066/.058
Experiment1.2 $NIR \rightarrow VIS$	59.11	.526	.900/.700	.062/.064

Table 4.4 Table captures Spectrum translation benchmark – concordance of the VIS FER model and the NIR FER model for the first 2 models; the last model is measuring the concordance between the original model and the original model with the NIR data translated to VIS data with the model from *Experiment1.2*. The concordance is measured on the images from the top-quarter of the OuluCasia image sets.

Chapter 5
Discussion

This last chapter before the conclusion discusses and interprets the results, datasets, and approaches used in this work. Also, the unfinished and future work is presented.

First of all, several observations suggest that the *Combined dataset* is problematic. The balance between valence and arousal in the *Combined dataset* is not the same as AffectNetNIR's, spatial and categorical performance on AffectNetNIR is surprisingly the same or better when trained on the Combined dataset (although the model was pretrained on VIS AffectNet, one could assume, that results would be worse when not pretrained on AffectNetNIR) and also there is a low concordance of test metrics when evaluated on the *Combined dataset* and *Combined dataset without the OuluCasia* images. There are findings in the spectrum translation benchmark section that extend this list as well. This might be because of inappropriate (automatic) annotations for both categorical and spatial labels in the OuluCasia dataset, low inner concordance, acting inadequacy or simply lack of sufficient data. Custom datasets (CustomDB and CustomMorphSet) extended the palette of true NIR images which differs from the other databases with its' human race distribution and sensor used for acquiring the NIR image (see different colour types of NIR image in 4.2). Other obtained datasets proved to be valuable. Even though the OuluCasia's and BUAA's VIS images were not used for spectrum translation, they were utilized in the benchmarks. Last but not least, as partially mentioned, all of the obtained NIR databases are from mostly lab conditions and people are not actors, who could show their mental state more expressively, which on the other hand provides AffectNet database. Thus those datasets including CustomDB should be taken a little less seriously when testing/training the FER model. Nevertheless, they still represent the true NIR images, which AffectNetNIR does not.

The evaluation of two proposed face detection methods, CenterFace and RetinaFace, was conducted as part of the assignment. These methods demonstrated exceptional detection capabilities in NIR images, with RetinaFace showing impressive results. On the other hand, CenterFace stood out for its balance between accuracy and inference speed. The models for both methods were sourced externally, with CenterFace derived from the original work and RetinaFace from a Python module. Given the assignment's requirement to propose two face detection methods, it was deemed unnecessary to develop another detector, considering it to be more than sufficient performance of the existing ones. Therefore, the emphasis was placed on implementing spectrum translation models. This approach not only fulfilled the assignment's requirements but also allowed for an efficient use of resources and focus on areas where improvement was needed.

Overall, $VIS \rightarrow NIR$ translation has satisfying results and it can be used in **Approach 2** (described in subsection 3.1) to transfer the FER AffectNet database to the NIR spectrum so it can be used for training the FER model (transferred database referred to as *AffectNetNIR*). On

the other hand, the $NIR \rightarrow VIS$ translation is more problematic, because the transfer is of a lower quality and generally harder as a task. All the fake faces are translated exactly as from the NIR image (they do not overfit) and all the faces resemble their NIR counterpart, only the parts of the faces are sometimes badly separated such that they blend in and sometimes faces have unnatural colour. Therefore, transferred images might be inadequately generated and subsequently, the FER prediction would be distorted. However, as the results of the **Experiment2.5** suggest, it served the purpose of recognizing the facial expression when employing **Approach 1**, though with a lower performance in the categorical metrics. As for the spectrum translation benchmark from $VIS \rightarrow NIR$, the concordance was considerable for both categorical and spatial predictors. Although the classifier's concordance appears low at 65%, it's quite significant considering that the state-of-the-art for FER is also 65% of accuracy. Similarly, the concordance of the spatial labels regressors is significant with a merely average distance of ≈ 0.5 when again considering the SOTA spatial labels prediction. The translation the other way around, $NIR \rightarrow VIS$, offers slightly lower concordance of categorical predictions with a decrease of 5%, and the regressor concordance is slightly lower as well – valence value is comparable and arousal value performs worse. Nevertheless, the decrease is surprisingly mild when considering the quality of fake VIS images.

As for the FER, overall, the *Experiment2.1.0* model outperforms or ties others in terms of normalized accuracy and F1 score on both the *Combined* dataset and AffectNetNIR. The Experiment2.1, on the other hand, excels in spatial predictions with the lowest normalized RMSE for the *combined* dataset and matches the performance of Experiment2.1.0 on AffectNetNIR. Those results again suggest that there is no problem with artificial NIR images in AffectNetNIR since the models trained on artificial data outperforms models trained on . While **Experiment2.1.0** seems to be the superior model overall, slightly surpassing Experiment2.1. The models trained on authentic NIR images, particularly Experiment2.3, also demonstrate commendable performance when considering issues of *Combined* dataset.

As for the FER, the experiments have been tested on two main test sets – AffectNetNIR and Combined dataset [1]. The best model on the AffectNetNIR was the one from the **Experiment2.1.0** on both categorical and spatial predictions slightly surpassing the models from the Experiment2.1 and the Original model with its normalized accuracy and F1 score being *0.585* and RMSE loss for spatial labels equal to *0.33* and *0.24* respectively. On the true NIR images, a Combined dataset, the best proved to be models trained on the Combined dataset itself. Model from the **Experiment2.2** achieved the highest normalized accuracy with 49% and F1 score of 0.48 surpassing by a margin of 2% the Experiment2.1.0 for accuracy. For the predictions of valence/arousal labels was best-suited **Experiment2.1** marginally ahead of Experiment2.3. Altogether, the best model appears to be the one from Experiment2.1.0.

The question of why the metrics on the Combined dataset are lower than the ones from AffectNetNIR might have several possible explanations. The reasons could be a qualitative difference between true and generated NIR images, an insufficient amount of data or inadequacies of the Combined dataset discussed earlier. As the answer is probably a combination of those, the author assesses the latter one to have the biggest impact mildly above the first-mentioned reason. However, more testing on high-quality annotated true NIR images would be necessary to prove that the model predicts comparably on true and artificial NIR images.

Finally, the whole system of acquisition of expressions from NIR images offers two fundamental approaches (described in section 3.1) from which is preferable Approach 2 – employing the model trained on the NIR images which yields better output than the first approach.

[1] When was tested models trained on Combined dataset, testing on the combined dataset was on the test set of the Combined dataset, not the whole dataset.

5.1 Comparison with State of the art

This work followed the approach proposed by Chen et al. (2022), which involved transforming the AffectNet database to the NIR spectrum. This makes Chen et al.'s study a key and only reference point, as it also concentrates on Facial Expression Recognition (FER) from NIR images. In Chen et al.'s work, the well-established OuluCasia database was used to train the CycleGAN model for the translation of the $VIS \rightarrow NIR$ spectrum. This thesis broadened their methodology by integrating additional databases, including images from AffectNet. This augmentation potentially enhances the quality of translations, particularly when generating VIS images, due to the increased volume and diversity of data. Furthermore, this research expanded upon their work by predicting categorical labels as well. The models developed in this research significantly outperformed those of Chen et al. Their RMSE losses for valence and arousal labels stood at 0.447 and 0.373 respectively, while this work achieved more favourable losses of 0.377 and 0.303 for valence and arousal respectively. The advancements could be attributed to the employment of a more potent architecture in MobileNet or a better transformation of AffectNet to the NIR spectrum.

When compared to the models trained on standard models trained on AffectNet in the visible spectrum, it achieves close state-of-the-art accuracy with 63.2% being close to 64.7%[2] and slightly better results compared with the *Original* model. However, those comparison makes sense only when artificial NIR images resemble true ones which was discussed earlier in this chapter.

5.2 Unfinished and future work

One of the possible improvements might be focusing more on $NIR \leftrightarrow VIS$ translation. First, the translation from $VIS \rightarrow NIR$ might be improved, however, in the current state, it is sufficient enough, since it serves only for translating the AffectNet dataset. A possible way to improve this might be through improving CycleGAN's architecture or seeking a better alternative.

The translation from $NIR \rightarrow VIS$ definitely might be improved. Liang et al. [37] (2024) propose the unpaired translation between RGB grayscale and VIS images with their *Conditional CycleGAN*, suitable for faces. Their Conditional CycleGAN stems from the idea of utilizing the YUV colour representation format.

The YUV[3] colour encoding system, widely used in analogue television, represents color as one luma (Y) and two chrominance (UV) components. The Y component corresponds to the brightness, while U and V carry colour information. This system was developed to allow colour television to be compatible with black-and-white infrastructure, with the Y component representing the black-and-white (effectively grayscale) signal and the UV signal providing the colour. Thus, CycleGAN could utilize this principle where there would be fewer channels. The generator transferring to VIS would then from grayscale image (Y channel) generate chrominance components (UV channels). The remaining generator then generates Y channels from UV, however, this translation would not be usable for NIR, since it does not equal grayscale. This architecture was implemented, and trained, but the results were stuck in local minima. However, insufficient time was taken for hyperparameter tuning and implementation, so this might be the focus of future experiments. Another idea for improving translation to VIS is inflating NIR data by adding a reasonable amount of images from AffectNet translated to NIR spectra. Since the translation images are quite believable, it could enhance the training since there are not enough NIR images of high diversity, which AffectNet provides. Nevertheless, this assumes that there is a first capable translator from $VIS \rightarrow VIS$.

Additionally, the CycleGAN can be improved. This Kaggle notebook[43] implements several improvements of standard CycleGAN and provides the usefulness of improvement in its use-

[2] When considered that the second

[3] Despite its name, YUV is often used to refer to colour spaces encoded using YCbCr.

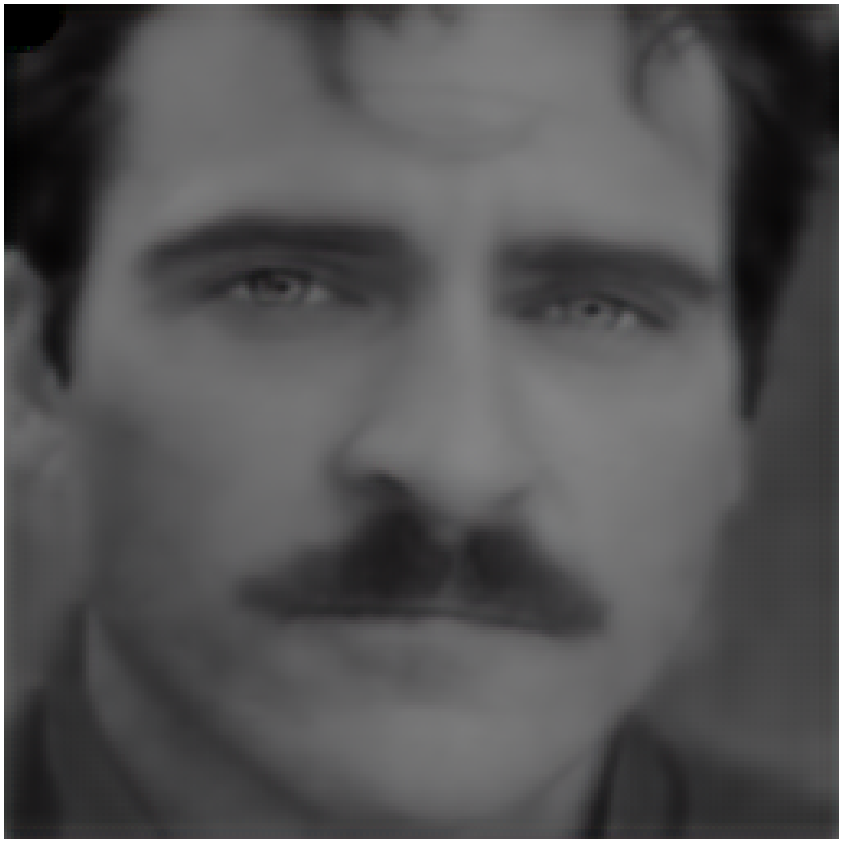


Figure 5.1 CycleGAN with attention mechanism delivered very promising results in the image above – provides smooth and accurate appearance resembling colouring of the NIR image. Although the resolution might still be improved, it should not be an issue.

case. The big enhancement of the model was implementing a transformer with residual blocks and introducing skip connections alongside others. Early experiments where those improvements were applied showed promising results, however, due to time limitations, the other experiments were not further explored. The demonstrative image is shown in the figure 5.1 – the image appears smooth and without "grid" as occurs in previous experiments and good colouring resembling the NIR image. Some parts of the image look of a low resolution or lower contrast, nevertheless, it has still space for improvement as it is from only the second epoch. Other options for possible improvements might be employing the FFE-CycleGAN [70] which utilizes pixel and feature loss for training a model on the paired BUAA dataset. This model would then translate between $VIS \leftrightarrow NIR$ spectra which would be later pretrained on unpaired NIR/VIS datasets without pixel loss.

Regarding the FER, the MorphSet dataset might be extended by morphing more images and annotated. Also, more time could be spent on the development of DDAMFN since it has a higher potential than MobileNet.

Chapter 6

Conclusion

This research primarily delves into the realm of facial expression recognition (FER), specifically focusing on images captured through a near-infrared (NIR) camera. The study also explores the methods employed to acquire these images. The research was inspired by previous work and by predicting not just categorical emotions, but also the valence arousal values of the Circumplex model of affect. This model offers a more nuanced description of an individual's current mental state. Two potential paths found in existing literature were considered: one involved training on a small, lab-limited dataset, while the other proposed transforming AffectNet, the most extensive available dataset, into NIR spectral images and training the model on that. The latter was chosen due to its potential robustness in real-world scenarios. Another second approach was also proposed, which involves detecting faces, translating them to the visible spectrum (VIS), and then applying a facial expression recognition model to ascertain a person's mental state. This whole work was then further divided into three subtasks: face detection, image spectrum translation, and facial expression analysis.

The first part of the research involved investigating two face detection methods: the pre-trained CenterFace, known for its balance between accuracy and inference time, and RetinaFace, renowned for its detection accuracy. Both detectors were tested on NIR images using a proposed benchmark and demonstrated near-perfect results.

The second part of the research focused on the method of transferring NIR images to VIS and vice versa. The CycleGAN architecture emerged as the most suitable method based on the available databases. These datasets were merged and extended with a custom dataset created from face morphing, known as the CustomMorphSet dataset. Subsequently, the CycleGAN model was trained and tested on this dataset.

Before the facial expression recognition phase, AffectNet was transferred into the NIR spectrum, yielding a dataset of artificial NIR images, termed AffectNetNIR. A novel dataset, CustomDB, was collected for verification and training purposes. It consists of approximately 200 images annotated with categorical emotions and valence arousal labels. An additional 600 images from other databases were annotated, and these were combined with the OuluCasia dataset to create a dataset of true annotated NIR images, called the Combined dataset.

The final part of the research involved studying a facial expression recognition system. Several variants of the MobileNet-based architecture network and the current state-of-the-art model for facial expression recognition, the DDAMFN network, were trained. Surprisingly, MobileNet trained on AffectNetNIR outperformed the original solution trained on VIS AffectNet and emerged as the better model with a near state-of-the-art solution that is held by the mentioned DDAMFN architecture. Testing on the Combined dataset yielded comparable results in the prediction of spatial labels but lower performance in the prediction of categorical labels. However, it was argued that this could be due to low inner concordance of the combined dataset.

Despite this, the research significantly surpassed the results of Chen et al. (2022), who introduced the transformation of AffectNetNIR into the NIR spectra.

After training the model for expression recognition, a spectrum translation benchmark was tested, comparing the concordance of MobileNet trained on VIS AffectNet and MobileNet trained on NIR AffectNet. The results were substantial for transfer to NIR and slightly less so for transfer to VIS. However, when considering the state-of-the-art accuracy in the FER task, the results were significant.

In summary, this research tested two models for face detection, trained a model for translating VIS to NIR images, a model for translating NIR to VIS images, and trained DDAMFN and MobileNet models, alongside other variants of these architectures.

For practical application, a user-friendly, customizable Python module was developed. This module encapsulates the entire pipeline, from acquiring a face from an image to retrieving and displaying the facial expression from it. The module can also process video as an input, yielding the same video with highlighted predictions for each face, allowing for easy assessment of expression acquisition quality.

Appendix A

Additional results

The table A.1 is an extended version of FER results 4.1 that contains all experiments and all test sets mentioned in the experiments section in chapter methods 3.5.2.

Table A.2 captures the results of metrics of the spectrum translation benchmark.

Measured was concordance of the NIR FER model and the VIS FER model on which this NIR FER model was pretrained on. Multiple NIR FER models were evaluated in concordance with the VIS model – models from the Experiment2.1.0 and the Experiment2.1. Also, the concordance was measured on multiple parts of the OuluCasia dataset – the whole dataset, the upper half of each set (upper half mages in a sequence of every emotion per patient set) and upper quarter of each set similarly. Measuring the concordance on those parts is important because images in the OuluCasia happen to be more ambiguous in the first half of the sequence, which is also visible in the results, especially in the categorical metrics.

Tracked metrics are percentage-wise concordance and Cohen's Kappa concordance metrics evaluating the similarity of predictions of categorical labels. Regarding the spatial labels, the CCC and the average distance between values serve this purpose. The Cohen's Kappa and CCC metrics are common metrics that evaluate the concordance of the models' predictions. Cohen's Kappa has the advantage over pure percentage concordance because it subtracts possible random matches.

Additionally, table A.3 captures the distribution of concordance (percentage-wise) by expression category. Benchmark is introduced in section 3.5.1.2 and results are discussed in the corresponding section in chapter *Results*.

In the figure A.1 is demonstrated $NIR \rightarrow VIS$ translation of images using the model from the **Experiment1.2**. Images are from AffectNetNIR translated with the model from **Experiment1.2**. The source images were not displayed because they match and do not show any signs of overfitting. Alongside the appealing images are the bad ones as well. However, those images are images from AffectNetNIR, thus artificial NIR images, that were not in the train set. Translation from true images depicts figure A.2 with the OuluCasia and BUAA images in the first row and the manually collected images for CustomDB in the rest of the rows. Images, including good and bad examples, are not as appealing as from the AffectNetNIR, and the quality varies a lot. The images from CustomDB generally generate better outcomes than those from OuluCasia. Another observation is that people with black hair elicit better outcomes; perhaps the network does not need to handle hair colour, since black hair is distinctible from NIR images.

id	model [test set]	categorical		spatial (valence/arousal)			normalized		
		top–ACC (top–1/2/3)	F1	RMSE	CCC	SAGR	ACC	F1	RMSE
(a)	DDAMFN [AffectNet][77] - *SOTA*	.647/ -/-	-	-/-	-/-	-/-	-	-	-/-
(b)	MobileNet [AffectNet][40] - *Original*	.620/ -/-	.470	.399/.298	.644/.411	.755/.660	.580	.580	.342/.319
(c)	MobileNet [Custom160][40] - *Original*	.606/ .775/.868	.549	.339/.399	.665/.477	.743/.650	.554	.537	.273/.321
(d)	EfficientNet-B0 [AffectNetNIR] [4] (2022)	-/ -/-	-	.447/.373	.527/.426	-/-	-	-	-/-
(e)	Experiment2.1.0 [AffectNetNIR]	.629/ .812/.902	.465	.377/.303	.693/.403	.752/.653	.585	.584	.333/.324
(f)	Experiment2.1.0 [combined dataset]	.504/ .711/.835	.442	.312/.308	.658/.361	.631/.604	.473	.449	.312/.353
(g)	Experiment2.1.0 [combined wo oulucasia]	.468/ .700/.817	.374	.359/.390	.658/.346	.642/.600	.414	.370	.313/.360
(h)	Experiment2.1 [AffectNetNIR]	.632/ .809/.900	.462	.376/.304	.695/.404	.753/.651	.573	.573	.332/.327
(i)	Experiment2.1 [combined dataset]	.504/ .704/.822	.433	.338/.356	.680/.424	.626/.607	.460	.420	.302/.342
(j)	Experiment2.1 [combined wo oulucasia]	.496/ .713/.825	.384	.345/.378	.679/.389	.626/.601	.416	.360	.298/.345
(k)	Experiment2.2 [AffectNetNIR]	.588/ .783/.887	.399	.425/.348	.516/.176	.717/.626	.469	.454	.375/.365
(l)	Experiment2.2 [combined dataset]	.536/ .715/.840	.480	.381/.400	.407/.155	.571/.595	.492	.483	.327/.340
(m)	Experiment2.2 [combined wo oulucasia]	.575/ .734/.867	.503	.364/.412	.441/.148	.592/.601	.513	.500	.309/.358
(n)	Experiment2.3 [AffectNetNIR]	.617/ .797/.895	.414	.429/.329	.519/.229	.716/.628	.492	.482	.368/.345
(o)	Experiment2.3 [combined dataset]	.540/ .727/.859	.464	.363/.388	.481/.199	.564/.595	.476	.448	.317/.332

Continued on next page

id	model [test set]	categorical		spatial (valence/arousal)			normalized		
		top–ACC (top–1/2/3)	F1	RMSE	CCC	SAGR	ACC	F1	RMSE
(p)	Experiment2.3 [combined wo oulucasia]	.495/ .716/.867	.418	.350/.400	.468/.164	.584/.601	.427	.401	.300/.346
(q)	Experiment2.4 [AffectNetNIR]	.619/ .828/.918	.455	.391/.300	.680/.418	.741/.662	.556	.555	.343/.326
(r)	Experiment2.4 [combined dataset]	.491/ .712/.820	.426	.360/.361	.647/.430	.625/.612	.438	.407	.317/.338
(s)	Experiment2.4 [combined wo oulucasia]	.503/ .727/.825	.383	.369/.374	.659/.420	.636/.622	.408	.350	.314/.335
(t)	Experiment2.5 [AffectNetNIR]	.631/ .823/.913	.456	.404/.282	.643/.392	.723/.647	.541	.542	.354/.317
(u)	Experiment2.5 [combined dataset]	.475/ .672/.798	.408	.352/.360	.637/.355	.619/.612	.420	.385	.308/.342
(v)	Experiment2.5 [combined wo oulucasia]	.453/ .670/.797	.329	.352/.376	.675/.334	.650/.608	.357	.305	.304/.339

Table A.1 This table is an extended version of table 4.1 (with the description) that captures FER experiments.

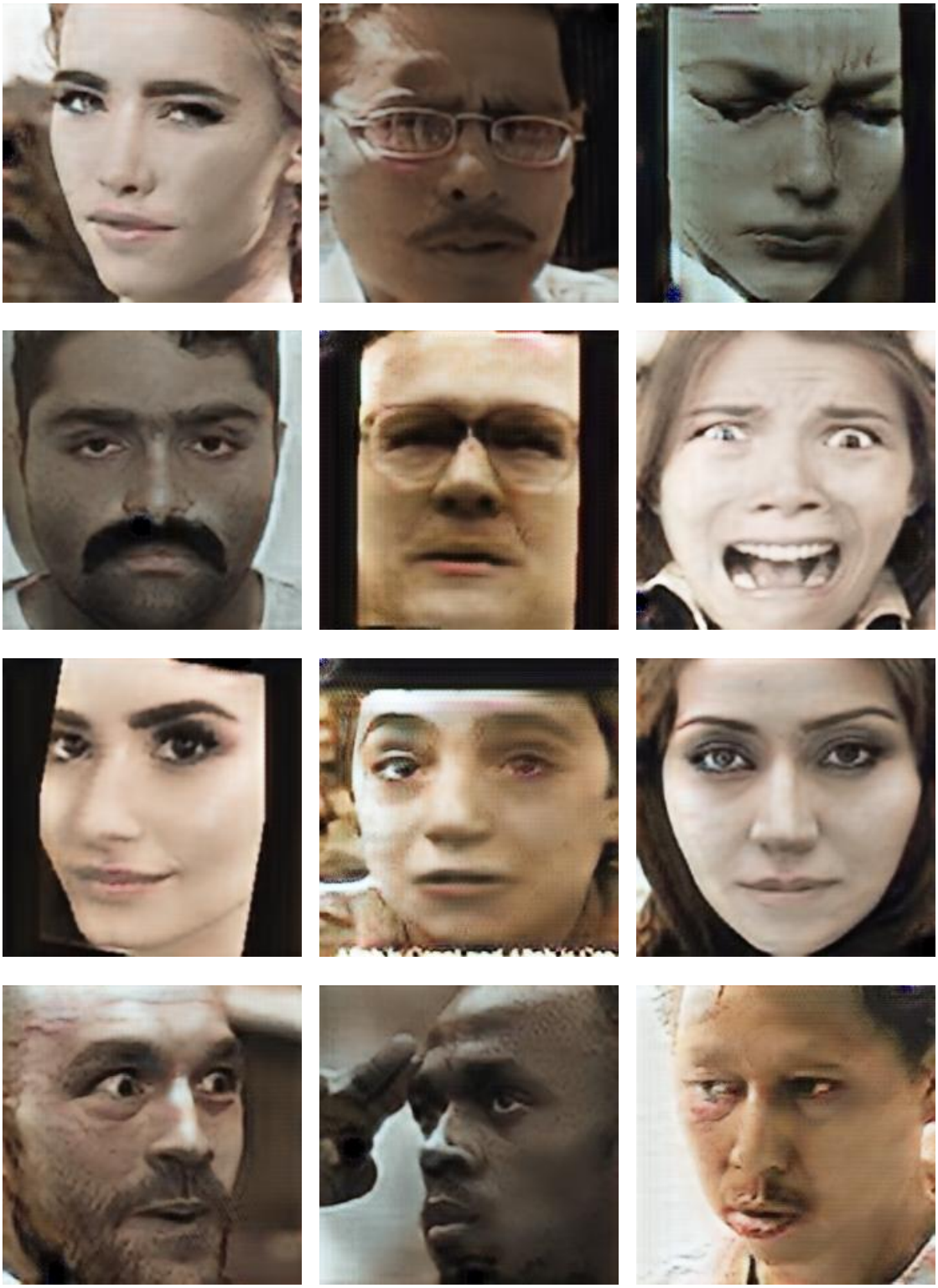


Figure A.1 A figure shows good and bad examples of $NIR \rightarrow VIS$ translation results from the **Experiment1.2**. The source images are from the AffectNetNIR image set translated from AffectNet with the **Experiment1.2** model, thus those source images are generated images. Those images were not in the train set.

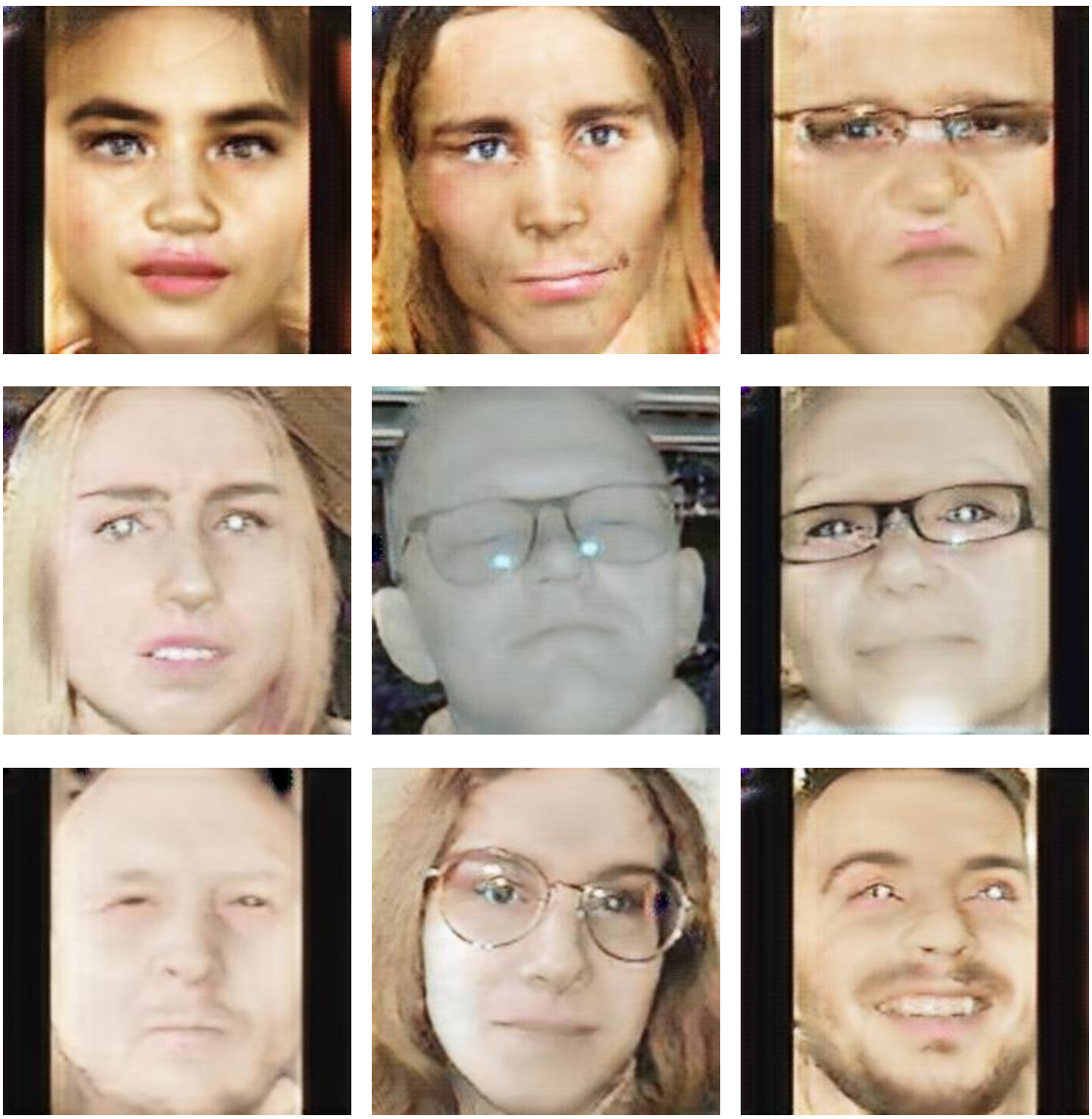


Figure A.2 A figure shows good and bad examples of $NIR \rightarrow VIS$ translation results from the **Experiment1.2**. The source images are from the *Combined dataset*, thus true NIR images. The first row contains images from BUAA and OuluCasia datasets. In the second and third rows are images collected in the CustomDB.

Part per experiment		Categorical metrics		valence/arousal metrics	
		Concordance	Cohen's Kappa	CCC	Avg. dist.
Experiment2.1.0	**All**	59.42	.523	.869/.729	.060/.061
	Top-half	63.28	.571	.909/.771	.059/.059
	Top-quarter	65.36	.595	.921/.800	.057/.056
Experiment2.1	**All**	59.326	.522	.839/.729	.069/.063
	Top-half	63.763	.576	.884/.772	.069/.061
	Top-quarter	65.448	.596	.895/.800	.066/.058
Experiment2.1.0 – RetinaFace	**All**	57.85	.502	.866/.734	.058/.060
	Top-half	62.02	.557	.902/.781	.059/.056
	Top-quarter	65.44	.597	.915/.811	.056/.053
Experiment1.2 $NIR \rightarrow VIS$	**All**	51.70	.438	.850/.629	.062/.065
	Top-half	56.58	.497	.887/.657	.064/.066
	Top-quarter	59.11	.526	.900/.700	.062/.064

Table A.2 The table presented below summarizes the results of the Spectrum translation benchmark. It shows the level of agreement between the VIS FER model and the NIR FER model for the first three models. The last model, on the other hand, measures the agreement between the original model and the original model with the NIR data translated to VIS data using the model from the *Experiment1.2*. The third model uses RetinaFace to detect faces, and each model includes all the subsets that were measured earlier.

Part per Experiment		Neutral	Happy	Sad	Fear	Surprise	Angry	Disgust
E2.1.0–ret	**All**	58%	68%	45%	43%	74%	55%	62%
E2.1.0–ret	**Top-half**	-	75%	45%	44%	82%	57%	68%
E2.1.0–ret	**Top-quarter**	-	81%	46%	51%	86%	58%	70%
E2.1.0	**All**	58%	69%	50%	51%	76%	53%	62%
E2.1.0	**Top-half**	-	75%	47%	53%	84%	54%	67%
E2.1.0	**Top-quarter**	-	79%	49%	53%	87%	56%	69%
E2.1	**All**	55%	70%	50%	55%	78%	51%	60%
E2.1	**Top-half**	-	75%	49%	55%	86%	53%	65%
E2.1	**Top-quarter**	-	77%	51%	55%	89%	54%	67%
E1.2–2VIS	**All**	47%	68%	49%	44%	64%	39%	57%
E1.2–2VIS	**Top-half**	-	72%	47%	47%	73%	40%	61%
E1.2–2VIS	**Top-quarter**	-	77%	50%	47%	77%	42%	62%

Table A.3 The table shows the percentage of agreement between VIS and NIR models based on different expressions. The abbreviation *E* stands for *Experiment* and *ret* indicates that the faces were detected by the RetinaFace detector. If *ret* is not present, it means that faces were detected by the CenterFace detector. Additionally, the *2VIS* suffix shows that NIR images were translated to the VIS spectrum using a specific spectrum translation model, and then predicted with the original FER model to determine their agreement with the original FER model on true VIS images.

Bibliography

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.

[2] Saeed Anwar, Muhammad Tahir, Chongyi Li, Ajmal Mian, Fahad Shahbaz Khan, and Abdul Wahab Muzaffar. Image colorization: A survey and dataset. *arXiv preprint arXiv:2008.10774*, 2020.

[3] John Bernhard, Jeremiah Barr, Kevin W Bowyer, and Patrick Flynn. Near-ir to visible light face matching: Effectiveness of pre-processing options for commercial matchers. In *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8. IEEE, 2015.

[4] Calvin Chen and Stefan Winkler. Generating near-infrared facial expression datasets with dimensional affect labels. *arXiv preprint arXiv:2206.13887*, 2022.

[5] Weijun Chen, Hongbo Huang, Shuai Peng, Changsheng Zhou, and Cuiping Zhang. Yolo-face: a real-time face detector. *The Visual Computer*, 37:805–813, 2021.

[6] Francois Chollet et al. Keras, 2015. URL: `https://github.com/fchollet/keras`.

[7] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641*, 2019.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[9] Hao Dou, Chen Chen, Xiyuan Hu, and Silong Peng. Asymmetric cyclegan for unpaired nir-to-rgb face image translation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1757–1761. IEEE, 2019.

[10] Shichuan Du, Yong Tao, and Aleix M Martinez. Compound facial expressions of emotion. *Proceedings of the national academy of sciences*, 111(15):E1454–E1462, 2014.

[11] Paul Ekman and Wallace V Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124, 1971.

[12] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[13] Ellen Goeleven, Rudi De Raedt, Lemke Leyman, and Bruno Verschuere. The karolinska directed emotional faces: a validation study. *Cognition and emotion*, 22(6):1094–1118, 2008.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[15] Ian J Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part III 20*, pages 117–124. Springer, 2013.

[16] Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE transactions on knowledge and data engineering*, 35(4):3313–3332, 2021.

[17] Sebastian Handrich, Laslo Dinges, Ayoub Al-Hamadi, Philipp Werner, and Zaher Al Aghbari. Simultaneous prediction of valence/arousal and emotions on affectnet, aff-wild and afew-va. *Procedia Computer Science*, 170:634–641, 2020.

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

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

[20] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.

[21] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[22] Fangzheng Huang, Xikai Tang, Chao Li, and Dayan Ban. Near-infrared and visible light face recognition: a comprehensive survey. *Soft Computing*, pages 1–20, 2023.

[23] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[24] Yongrui Huang, Jianhao Yang, Siyu Liu, and Jiahui Pan. Combining facial expressions and electroencephalography to enhance emotion recognition. *Future Internet*, 11(5):105, 2019.

[25] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. arxiv e-prints. *arXiv preprint arXiv:1611.07004*, 2016.

[26] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[27] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two pure transformers can make one strong gan, and that can scale up. *Advances in Neural Information Processing Systems*, 34:14745–14758, 2021.

[28] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[29] Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009.

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[31] Byoung Chul Ko. A brief review of facial emotion recognition based on visual information. *sensors*, 18(2):401, 2018.

[32] Dimitrios Kollias and Stefanos Zafeiriou. Expression, affect, action unit recognition: Aff-wild2, multi-task learning and arcface. *arXiv preprint arXiv:1910.04855*, 2019.

[33] Oliver Langner, Ron Dotsch, Gijsbert Bijlstra, Daniel HJ Wigboldus, Skyler T Hawk, and AD Van Knippenberg. Presentation and validation of the radboud faces database. *Cognition and emotion*, 24(8):1377–1388, 2010.

[34] Kuan-Ting Lee, En-Rwei Liu, Jar-Ferr Yang, and Li Hong. An image-guided network for depth edge enhancement. *EURASIP Journal on Image and Video Processing*, 2022(1):6, 2022.

[35] Shan Li, Weihong Deng, and JunPing Du. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2852–2861, 2017.

[36] Stan Li, Dong Yi, Zhen Lei, and Shengcai Liao. The casia nir-vis 2.0 face database. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 348–353, 2013.

[37] Chen Liang, Yunchen Sheng, and Yichen Mo. Grayscale image colorization with gan and cyclegan in different image domain. *arXiv preprint arXiv:2401.11425*, 2024.

[38] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *Advances in neural information processing systems*, 30, 2017.

[39] Jiawei Mao, Rui Xu, Xuesong Yin, Yuanqi Chang, Binling Nie, and Aibin Huang. Poster++: A simpler and stronger facial expression recognition network. *arXiv preprint arXiv:2301.12149*, 2023.

[40] Vadlejch Martin. Analỳza vỳrazu tváře z obrazu v reálném čase. B.S. thesis, České vysoké učení technické v Praze. Vypočetní a informační centrum., 2021.

[41] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282, 2012.

[42] Ali Mollahosseini, Behzad Hasani, and Mohammad H Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31, 2017.

[43] Dimitre Oliveira. Improving cyclegan monet paintings. `https://www.kaggle.com/code/dimitreoliveira/improving-cyclegan-monet-paintings/notebook`, 2024. Accessed: 2024-01-06.

[44] Michal Olszanowski, Grzegorz Pochwatko, Krzysztof Kuklinski, Michal Scibor-Rylski, Peter Lewinski, and Rafal K Ohme. Warsaw set of emotional facial expression pictures: a validation study of facial display photographs. *Frontiers in psychology*, 5:1516, 2015.

[45] Rafael Padilla, CFF Costa Filho, and MGF Costa. Evaluation of haar cascade classifiers designed for face detection. *World Academy of Science, Engineering and Technology*, 64:362–365, 2012.

[46] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[47] Yuan-Yuan Pu, Colm O'Donnell, John T Tobin, and Norah O'Shea. Review of near-infrared spectroscopy as a process analytical technology for real-time product monitoring in dairy processing. *International Dairy Journal*, 103:104623, 2020.

[48] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[49] Rafael Redondo. FaceMorph, 12 2018. URL: `https://github.com/valillon/FaceMorph`.

[50] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[51] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[52] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[53] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[54] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

[55] Dr Priti Sadaria, Dr Haresh Khachariya, and Dr Jignesh Hirpara. Exploring the diverse applications of deep learning across multiple domains. *rrrj*, 2(1):183–200, 2023.

[56] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022.

[57] Nilu R Salim, Srinath V, Umarani Jayaraman, and Phalguni Gupta. Recognition in the near infrared spectrum for face, gender and facial expressions. *Multimedia Tools and Applications*, 81(3):4143–4162, 2022.

[58] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[59] Edgar Schonfeld, Bernt Schiele, and Anna Khoreva. A u-net based discriminator for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8207–8216, 2020.

[60] Sefik Ilkin Serengil and Alper Ozpinar. Lightface: A hybrid deep face recognition framework. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 23–27. IEEE, 2020. doi:10.1109/ASYU50717.2020.9259802.

[61] Athreya V Shet, BS Chinmay, Anvithkumar A Shetty, T Shankar, R Hemavathy, and P Ramakanth. Face detection and recognition in near infra-red image. In *2022 6th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, pages 1–6. IEEE, 2022.

[62] simon20010923. Ddamfn. https://github.com/simon20010923/DDAMFN, 2024.

[63] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[64] PA Srudeep. An overview on mobilenet: an efficient mobile vision cnn. *URL https://medium.com/@ godeep48/an-overview-on-mobilenet-an-efficient-mobile-vision-cnn-f301141db94d*, 2020.

[65] Matti Taini, Guoying Zhao, Stan Z Li, and Matti Pietikainen. Facial expression recognition from near-infrared video sequences. In *2008 19th international conference on pattern recognition*, pages 1–4. IEEE, 2008.

[66] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[67] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001.

[68] Vassilios Vonikakis, Neo Yuan Rong Dexter, and Stefan Winkler. Morphset: Augmenting categorical emotion datasets with dimensional affect labels using face morphing. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 2713–2717. IEEE, 2021.

[69] Vassilios Vonikakis and Stefan Winkler. Efficient facial expression analysis for dimensional affect recognition using geometric features. *arXiv preprint arXiv:2106.07817*, 2021.

[70] Huijiao Wang, Haijian Zhang, Lei Yu, and Xulei Yang. Facial feature embedded cyclegan for vis–nir translation. *Multidimensional Systems and Signal Processing*, pages 1–24, 2023.

[71] Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. *Advances in neural information processing systems*, 30, 2017.

[72] Yuxuan Xiao, Aiwen Jiang, Changhong Liu, and Mingwen Wang. Single image colorization via modified cyclegan. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 3247–3251. IEEE, 2019.

[73] Yuanyuan Xu, Wan Yan, Genke Yang, Jiliang Luo, Tao Li, and Jianan He. Centerface: joint face detection and alignment using face as point. *Scientific Programming*, 2020:1–8, 2020.

[74] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5525–5533, 2016.

[75] Aijing Yu, Haoxue Wu, Huaibo Huang, Zhen Lei, and Ran He. Lamp-hq: A large-scale multi-pose high-quality database and benchmark for nir-vis face recognition. *International Journal of Computer Vision*, 129(5):1467–1483, 2021.

[76] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016.

[77] Saining Zhang, Yuhang Zhang, Ye Zhang, Yufei Wang, and Zhigang Song. A dual-direction attention mixed feature network for facial expression recognition. *Electronics*, 12(17):3595, 2023.

[78] Guoying Zhao, Xiaohua Huang, Matti Taini, Stan Z Li, and Matti PietikäInen. Facial expression recognition from near-infrared videos. *Image and vision computing*, 29(9):607–619, 2011.

[79] Guoying Zhao, Xiaohua Huang, Matti Taini, Stan Z. Li, and Matti Pietikäinen. Facial expression recognition from near-infrared videos. *Image and Vision Computing*, 29(9):607–619, 2011. URL: `https://www.sciencedirect.com/science/article/pii/S0262885611000515`, `doi:10.1016/j.imavis.2011.07.002`.

[80] Enting Zhou, You Zhang, and Zhiyao Duan. Learning arousal-valence representation from categorical emotion labels of speech. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12126–12130. IEEE, 2024.

[81] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[82] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018.