INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Tecnologia e Gestão

Facial Expression Recognition under Partial Occlusion

Ana Sofia Figueiredo Rodrigues - a41737

Thesis presented to the School of Technology and Management in the scope of the Master in Informatics.

Supervisors:

Prof. Rui Pedro Lopes

Bragança

2023-2024

Facial Expression Recognition under Partial Occlusion

Ana Sofia Figueiredo Rodrigues - a41737

Thesis presented to the School of Technology and Management in the scope of the Master in Informatics.

Supervisors:

Prof. Rui Pedro Lopes

Bragança

2023-2024

Dedication

This work is dedicated to my family. Their unwavering support and encouragement have been the foundation of my journey through my master's degree. In times when motivation was hard to come by, they stood by me, offering strength and reassurance. Without their presence, patience, and belief in my potential, this achievement would not have been possible. To my family, thank you for being my constant source of inspiration and resilience.

Acknowledgment

I would like to express my deepest gratitude to my supervisor, Professor Rui Pedro Lopes, for his invaluable support, guidance, availability, and patience throughout this journey. His insights and suggestions were crucial in bringing this work to fruition.

I am also profoundly thankful to Júlio Castro Lopes, whose support and guidance were unwavering, even from a distance. His flexibility, availability, and thoughtful suggestions were instrumental in the development of this work.

Lastly, to my friends, my second family, thank you for being my steadfast source of encouragement, support, and joy. Your friendship has been a vital part of this journey.

Abstract

Facial expressions play a crucial role in conveying emotions, accounting for 55% of communication. Although humans naturally perceive these expressions, individual differences can make this recognition complex. Technological advancements seek to automate the identification of facial expressions, thereby improving interactions. Nonetheless, the obstruction of facial features caused by elements such as hand movements or hair presents substantial obstacles, complicating the precise recognition of expressions. This study investigates the impact of partial occlusion on facial expression recognition, specifically examining how occlusions from masks and Virtual Reality goggles affect model performance on the FERPlus and FERV39K datasets. The results reveal that occlusion reduces the accuracy of all models. Notably, the performance of EfficientNetB1 drops significantly from 92.9% to 74% when the mouth is obscured, in happiness, in FERPlus dataset, while ResNet18 performs poorest in recognizing fear, plummeting to 30% with eyes occlusion. In the FERV39K dataset, occlusion scenarios have a substantial effect on the accuracy of the neutral class. For example, in VGG19, the accuracy decreases sharply from 94.4% to 31.7% in the goggles occlusion scenario and to 30.4% in the mask occlusion scenario.

However, a three-class grouping enhances the overall performance, illustrated by the results obtained in the three models in both datasets, indicating the effectiveness of the approach in difficult situations. These findings emphasize the significant challenges occlusion poses for emotion recognition systems, highlighting the need for continued research in this field.

Keywords: Facial Expression Recognition, Partial Occlusion, Class Grouping

Resumo

As expressões faciais desempenham um papel crucial na transmissão de emoções, representando 55% da comunicação. Embora os humanos percebam essas expressões de forma natural, as diferenças individuais podem tornar esse reconhecimento complexo. Avanços tecnológicos procuram automatizar a identificação das expressões faciais, melhorando assim a interação humano-máquina ou outras aplicações. No entanto, a obstrução de características faciais causada por elementos como movimentos das mãos ou cabelo apresenta obstáculos substanciais, complicando o reconhecimento preciso das expressões. Este estudo investiga o impacto da oclusão parcial no reconhecimento de expressões faciais, examinando especificamente como as oclusões causadas por máscaras e óculos de Realidade Virtual afetam o desempenho dos modelos nos conjuntos de dados FERPlus e FERV39K. Os resultados revelam que a oclusão reduz significativamente a precisão de todos os modelos. Notavelmente, o desempenho do EfficientNetB1 cai significativamente de 92,9% para 74%, quando a boca é obscurecida, em *happiness*, no conjunto de dados FERPlus, enquanto o ResNet18 tem o pior desempenho no reconhecimento de *fear*, caindo para 30% com a oclusão dos olhos. No conjunto de dados FERV39K, os cenários de oclusão têm um impacto substancial na precisão da classe *neutral*. Por exemplo, no VGG19, a *accuracy* diminui drasticamente de 94,4% para 31,7% no cenário de oclusão com óculos e para 30,4% no cenário de oclusão com máscara.

No entanto, um agrupamento de três classes melhora o desempenho geral, como ilustrado pelos resultados obtidos nos três modelos em ambos os conjuntos de dados, indicando a eficácia da abordagem em situações difíceis. Esses resultados ressaltam os desafios significativos que a oclusão impõe para os sistemas de reconhecimento de emoções,

destacando a necessidade de pesquisas contínuas nesta área.

Palavras-chave: Reconhecimento de Expressão Facial, Oclusão Parcial, Agrupamento de Classes

Contents

List of Figures

Acronyms

ACNN Attention Convolutional Neural Network. 16

AEs Auto-Encoderes. 6

AI Artificial Intelligence. 5–7

BN Batch Normalization. 10

CC Correct Classification. 10

CFD Compact Face Descriptor. 15

CFD-OD Compact Face Descriptor with Occlusion Detection. 15

CFD-OD-WL Compact Face Descriptor with Occlusion Detection and Weight Learning. 15

CNN Convolutional Neural Network. 10, 19

CNNs Convolutional Neural Networks. 6, 25, 27

CPU Central Processing Unit. 40

CSM Co-Saliency Model. 15

CV Computer Vision. 5–7, 25

DL Deep Learning. 8, 36

DNMF Discriminative Nonnegative Matrix Factorization. 15

EEM Emotional Education Mechanism. 12

ESTIG Escola Superior de Tecnologia e Gestão. 33

F-LGBPHS Fused Local Gabor Binary Pattern Histogram Sequence. 15, 16

FACS The Facial Action Coding System. 9

FER Facial Expression Recognition. 1, 5, 10, 19

FSE Facial Soft Biometrics Estimation. 15

GLCM Grey-Level Co-occurrence Matrix. 5, 15, 16

GMMs Gaussian Mixture Models. 15

GPU Graphics Processing Unit. 33, 40

GSNMF Graph-Structured Nonnegative Matrix Factorization. 15

HOG Histogram of Oriented Gradients. 5, 15

KNN K-Nearest Neighbors. 15, 16

KTN Knowledgeable Teacher Network. 12

LBP Local Binary Patterns. 5, 15, 16

LGBPHS Local Gabor Binary Pattern Histogram Sequence. 15, 16

M-KDDI-FER Multi-Label KDDI Facial Expression Recognition. 16

M-LFW-FER Multi-Label Face in the Wild for Facial Expression Recognition. 16

M-LGBPHS Multi-Scale Local Gabor Binary Pattern Histogram Sequence. 15, 16

MBConv Mobile Inverted Bottleneck Convolution. 30

MCC Matthews Correlation Coefficient. 15

ML Machine Learning. 6, 7

MSE Mean Squared Error. 16

MTCNN Multitask Cascade Convolutional Neural Network. 13, 15, 22

NLP Natural Language Processing. 7

NMS Non-Maximum Suppression. 22

NN Neural Network. 7

NNs Neural Networks. 7

O-Net Output Network. 23

OADN Occlusion-Aware Deep Network. 16

P-Net Proposal Network. 22

PSO Particle Swarm Optimization. 13

R-Net Refine Network. 23

RAN Region Attention Network. 16

RBF Radial Basis Function. 10

RBMs Restricted Boltzmann Machines. 6

ReLU Rectified Linear Unit. 29

ReLu Rectified Linear Unit. 10

ResNet18 Residual Network 18. 29

SAX Symbolic Aggregate approXimation. 17

SENet Squeeze-and-Excitation Network. 30

SIFT Scale-Invariant Feature Transform. 5

SNMF Sparse Nonnegative Matrix Factorization. 15

STLBP Spatio-Temporal Local Binary Pattern. 15

STSN Self-Taught Student Network. 12

SVM Support Vector Machine. 10, 13, 15

TFD Toronto Face Database. 10

TLU Threshold Logic Unit. 8

VGG Visual Geometric Group. 16, 29

VGG19 Visual Geometric Group 19. 29

VPN Virtual Private Network. 33

VR Virtual Reality. 2, 15, 45

Chapter 1

Introduction

Facial expression is one of the most intuitive, comprehensive, and effective ways to communicate inner emotions, playing a fundamental role in human daily activities. As an important component of intercultural communication, facial expressions are a dynamic channel of nonverbal communication that can convey both involuntary reactions and intentional gestures [1]. According to Mehrabian [2], communication between individuals is composed of 7% through writing, 38% through conversation, and a significant 55% through facial expressions.

Humans have developed the ability to read others' facial expressions instinctively, often recognizing them even when the person is unaware of their actions. For instance, features such as drawn-down eyebrows, contracted eyelids, and outward-drawn eyes are commonly associated with sadness. However, it is important to note that the same emotion can manifest differently in different individuals and contexts [3]. Extending this possibility to computers, the automatic identification of human facial expressions introduces several benefits. It has the potential to develop better and more useful human-computer interaction, provide visually impaired with haptic clues regarding the expression of others [4], monitor the motivation of students in the classroom [5], among many other applications. Even though challenging, it is constantly evolving with several proposed solutions by the scientific community [6].

Although Facial Expression Recognition (FER) has made significant improvements,

there are still significant practical challenges that prevent an accurate analysis. Partial occlusion is one of these, which means covering parts of facial features [7]. It can manifest itself in various scenarios, including instances where facial features, such as hand or head movements, temporarily obstruct part of the face or when facial components, such as hair or scarves, cause temporary blockages [8]. The complexity of extracting distinctive features from occluded facial areas is compounded by the presence of occlusion, leading to difficulties such as inaccurate feature localization, imprecise facial alignment, or errors in facial registration [7]. Some occlusion parameters that pose challenges for facial expression recognition include eye or mouth occlusion, where expressions involving eye movements or changes, such as surprise or happiness, may be difficult to recognize. Similarly, partial or self-occlusion, such as masks or sunglasses, can obscure relevant facial features, making expression recognition challenging. Addressing these occlusion parameters is essential for enhancing the effectiveness of facial expression recognition systems.

1.1 Objectives

The objective of this study is to investigate the impact of occlusion on facial expression recognition, with a particular focus on the challenges introduced by partial occlusions. Specifically, the study will examine how lower and upper occlusions — such as those caused by masks (as commonly worn during the COVID-19 pandemic) and Virtual Reality (VR) goggles affect the accuracy and reliability of facial expression recognition systems. Furthermore, the study will investigate how variations in facial expressions, such as happiness, sadness, anger, and surprise, interact with these occlusions, potentially leading to an impact on recognition performance.

1.2 Structure of the Document

The remainder of this dissertation is organized into five chapters, as follows: Chapter 1 provides an overview of facial expression recognition, discussing the challenges, significance and the objectives of this research. The field of facial expression recognition (FER) is reviewed in Chapter 2, which focuses on recent, relevant approaches and highlights various methods and their outcomes, as well as key terminologies. The method used to recognize facial expressions, both with and without masks and goggles, is detailed in Chapter 3, which involves class reduction, datasets, and network architectures. The results obtained through the applied methodology are presented in Chapter 4. Chapter 5 also presents its conclusions and suggests future research directions.

Chapter 2

Facial Expression Recognition

FER is a crucial technology in the fields of Computer Vision (CV) and Artificial Intelligence (AI). It involves the detection and categorization of facial expressions from images or videos to interpret human emotions. This section provides key terminology used in FER and explores the latest advances and methodologies in the state of the art.

2.1 Image Classification

Computer Vision is a field that focuses on enabling computers to understand and interpret digital images, such as photographs and videos. In CV, image classification is a crucial and fundamental task that involves categorizing images into predefined groups based on their content. Computer vision technology has greatly benefited from advancements in this task, making it essential for a wide range of applications. Image classification involves extracting key features from images to reduce dimensionality and enhance performance, using methods such as Grey-Level Co-occurrence Matrix (GLCM), Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), and Scale-Invariant Feature Transform (SIFT). Without manual intervention, feature learning automates the discovery of effective data representations, leading to improved classification accuracy. After identifying the most effective features, a classifier is instructed to classify new images. Extensive data and complex models are often required to automate this process with advanced techniques,

such as Convolutional Neural Networks (CNNs), Auto-Encoderes (AEs), and Restricted Boltzmann Machines (RBMs) [9].

Computer vision and pattern recognition have a specialized area called FER that is focused on automating the identification and analysis of facial expressions [10]. Throughout the years, FER is currently being applied in many different fields, such as human-computer interaction [11], healthcare [12], education [13] and many more [14].

Within the wider realm of AI, FER showcases the capabilities of Machine Learning (ML) and CV technologies in examining human emotions. This highlights their combined ability to deepen comprehension and interpretation of intricate emotional signals while underlining their effectiveness in solving complex problems and making intelligent decisions.

2.2 Machine Learning

AI refers to the field of science and engineering that focuses on reproducing, improving, and supplementing human intelligence by using artificial methods and technologies to create intelligent machines [15]. The ability to solve problems, discuss ideas, plan, write computer programs, drive cars, and cycle requires human intelligence. AI can be demonstrated by machines capable of successfully performing these tasks [16]. This domain has a wide range of applications, including speech recognition and image processing, natural language processing, smart robots, autonomous vehicles, energy systems, and healthcare [17]. Especially in healthcare, AI reveals several positive impacts, such as fast and accurate diagnostics, where programs like IBM Watson program [18] quickly analyze medical data to diagnose conditions and suggest treatment options. This area also contributes to therapeutic robots that provide companionship and assistance to elderly people, reducing anxiety and improving their quality of life. There are also AI-assisted surgeries, such as those using the Da Vinci system [19], that offer greater precision and less invasiveness, resulting in less trauma and faster recovery. Advances in AI radiology, including the development of new algorithms, improve the detection and analysis of diseases through

scans. In addition, AI enables remote diagnosis and consultation through virtual presence, allowing specialists to assist patients without the need for travel [20].

The ML process involves programming computers to improve their performance using example data or past experiences to optimize specific parameters. The field focuses on developing computer programs that automatically improve with experience [21]. In order for a system to function as AI, ML is only one component required: it allows AI to adjust to unexpected situations, identify patterns in diverse data sources, develop new behaviors from these patterns, and make decisions based on the outcomes of these behaviors. This field involves manipulating data through the use of algorithms. Moreover, the data must be suitable for analysis using the chosen algorithm or carefully prepared [22]. The different application domains of ML include CV prediction, semantic analysis, Natural Language Processing (NLP), and information retrieval. This field is utilized in CV for object recognition, detection, and processing. For prediction, subdomains such as classification, analysis, and recommendation are prevalent, with successful implementations in text classification, document classification, image analysis, medical diagnosis, prediction of network intrusion detection, and prediction of denial of service attacks. The purpose of semantic analysis is to connect the syntactic structures of paragraphs, sentences, and words to the level of writing as a whole. The focus of natural language processing is on programming computers to process natural language data correctly. Searching for information within a document, searching for documents, searching for metadata that describes data, as well as searching for databases of sounds and images, all are part of information retrieval [23]. Within the field of ML, one of the most promising and powerful techniques is the use of Neural Networks (NNs). A Neural Network (NN) is a system composed of interconnected processing elements, units, or nodes that mimic the functionality of animal neurons (Figure 2.1). The process capability of the network is encapsulated in the connection strengths, or weights, between units that are formed through a process of adaptation or learning from a set of training patterns. The structural and functional similarities between the human brain and artificial neurons are the reason for their relationship. The behavior of biological neurons in the brain can be mimicked by artificial neurons or nodes

in a neural network. Synapses, which are represented by weights, make adjustments to the strength of inputs before sending them to the equivalent cell body, where they combine signals to trigger activation. If activation goes above a certain threshold, the artificial neuron will output a high value, just like a biological neuron will fire an action potential when stimulated sufficiently. In the human brain, neurons use the basic processing and decision-making functions portrayed by Threshold Logic Unit (TLU), which is a simplified model. The term 'network' refers to interconnected artificial neurons, which can range from a simple single node to complex layered structures, similar to the interconnected networks of neurons in the brain responsible for processing and transmitting information [24], [25].

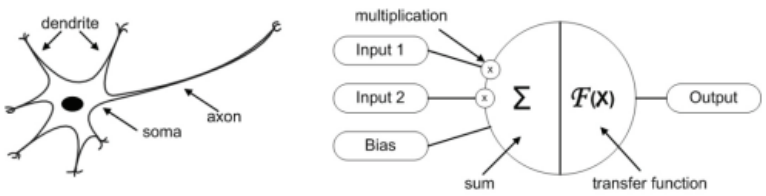


Figure 2.1: Biological and artificial neuron [25].

Deep Learning (DL) is a category of ML algorithms characterized by the use of multiple layers of computational units. Each layer independently acquires a unique representation of the input data, which is combined by subsequent layers in a hierarchical way [26]. This field is particularly appropriate for situations where the data is both highly complex and large in volume. The algorithms of this area do not follow linear paths like traditional learning algorithms but rather a hierarchical structure with increasing complexity and abstraction. The hierarchical structure applies non-linear transformations to input data and uses its learning to produce statistical models as outputs. The outputs are subjected to iterations until they reach an acceptable level of accuracy. Differentiating between DL and ML is essential, as ML uses algorithms to analyze data, learn from it, and make decisions that are similar to human reasoning. DL learns representations from data (such as images, videos, or text) automatically without the need for manually coded rules or human domain expertise. In practical terms, consider the task of detecting oranges

on a production line: traditional methods might require explicit rules like 'oranges are round' and 'oranges are orange' to guide machine learning, whereas DL autonomously identifies these characteristics without explicit instructions. This domain is now widely applied across various areas, such as analyzing online text conversations on platforms like Facebook and employed by tech giants such as Google, Baidu, and Microsoft for tasks such as image search and automated translation. It is incorporated into modern smartphones for tasks such as voice recognition (for example, Alexa and Siri), facial recognition (Face ID), and biometric security. The processing of medical images, including X-rays, requires DL, and in autonomous vehicles, it is crucial for mapping, environmental perception, and driver state detection [27].

2.3 Facial Expression Recognition

Paul Ekman and Wallace V. Friesen elucidated the universally acknowledged nature of the following seven expressions: happy, sad, surprised, fearful, angry, disgusted, and contemptuous [28]. Coined as the 'Universal Facial Expressions of Emotion', they also introduced a widely utilized metric system called The Facial Action Coding System (FACS), designed to capture facial actions relevant to these expressions. Some researchers have considered incorporating the neutral face as an additional expression for classification [29]–[31].

Although important for human interaction, most of these are not relevant for computer applications, so some of them are merged by similarity. The scientific literature provides an indication of how to perform this grouping, allowing the evaluation, effectiveness, and efficacy of the current state of the art. In addition to class grouping, our study also focuses on the issue of partial occlusion, a common occurrence in our daily lives due to factors such as masks worn during the COVID-19 pandemic, glasses or scarves, which can obscure significant parts of the face, and complicate the accurate recognition of emotional expressions.

The potential of FER is to revolutionize how machines interpret and respond to human emotional states, which has explored its different methodologies and applications.

Loizou [32] proposed and evaluated a system that analyzes speech and image signals for the 7 Universal Facial Expressions of Emotion. More than 70,000 people aged 20 to 74 years were recorded to produce voice and image recordings. Multiclassification models were used to identify the characteristics that distinguish these seven emotions, employing a Support Vector Machine (SVM) with 10-fold validation and a Radial Basis Function (RBF) kernel with parameters c=1 and $gamma$=0.01 [33]. The author achieved a Correct Classification (CC) score of 93%.

Devries et al. [34] proved that a system that is trained to understand facial geometry and recognize expressions is superior to a model that is only trained to recognize expressions. They decided on Zhu and Ramanan's facial landmark detector [35], which gives coordinates for 68 facial landmarks in each face, which delineate features such as the mouth, nose, eyes, and eyebrows. By focusing on the eyebrows and mouth, they represented every position with a binary mask image. They used a Convolutional Neural Network (CNN) architecture inspired by the winning model of the 2013 ICML Facial Expression Recognition Challenge [36] and had three fully connected convolutional layers with ReLU and max pooling. The output included three binary maps (one per landmark) modeling location and shape. The study used the ICML dataset [37] and the Toronto Face Database (TFD) [38]. The authors achieved an accuracy of 67.21% in the ICML dataset and 85.13% in the TFD dataset.

Li et al. [39] developed a method for capturing features using the ResNet50 deep residual network combined with a CNN. To improve the convergence of the model, Batch Normalization (BN) and Rectified Linear Unit (ReLu) were used in conjunction with CNN for the extraction of features. To streamline the experimental process and ensure result comparability, a new dataset was created in which a photographer captured facial expressions of 20 subjects, diverse in age and career, ten times each using a digital camera, resulting in 700 images that encompass seven types of facial emotions: happy, sad, fear, anger, surprise, disgust, and neutral. The authors obtained a result of 95.39% of overall accuracy.

Nan et al. [40] developed a lightweight A-MobileNet model for FER. The model was

created using an attention module in MobileNetV1 to improve local feature extraction of facial expressions, and dropout regularization was used to prevent overfitting. In addition, a combination of softmax loss and center loss is utilized to optimize model parameters, with the aim of decreasing the distance between classes and increasing the separation between classes. Experimental analyses were performed on two datasets: FERPlus [41] and RAF-DB [42]. The model achieved an accuracy rate of 84.49% on the RAF-DB dataset, which was followed by an accuracy rate of 88.11% on FERPlus.

2.3.1 Reduced Class Division for FER

There are two popular methods for reducing the number of classes in a dataset: using only some of the classes and ignoring the others, or grouping the classes into major groups (positive, negative, and neutral) (Figure 2.2), although there are studies that consider only two major groups (positive and negative), considering neutral as a negative expression [43]. It should be noted that this division does not strictly aim at improving the accuracy of classification methods, but rather for specific applications where this division is preferred [44]. However, as a general rule, classification accuracy is improved with fewer classes [45].

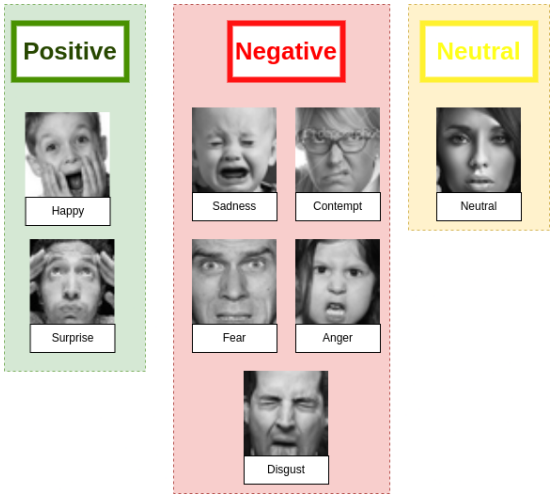


Figure 2.2: Example of class grouping into 3 major groups: positive, negative and neutral

Mozaffari et al. [29] developed a CNN to classify seven facial expressions in the FER-2013 dataset [46]: anger, disgust, fear, happiness, sadness, surprise, and neutral; they also divided the dataset, classifying only the top three emotions (happiness, fear, and anger, in this case). The authors normalized the images of this dataset and used data augmentation techniques, including rotating, shearing, and vertically and horizontally rotating the image to balance the dataset. They used three different models in their paper: a CNN model proposed by them and 2 pretrained VGG16 [47] and EfficientNet [48]. Their model achieved an accuracy of 72% for the 7-class classification and an accuracy of 86% for the 3-class classification, with the EfficientNet.

Other works also discard the other emotions and just maintain the classes that they considered the most valuable for their application. Morshed et al. [49] created a customer service system that evaluates the spontaneous facial expression of the client when they observe certain products. To classify facial expression, the authors trained, using the FER-2013 dataset [46], a CNN that recognizes three types of expressions: happy, sad, and neutral. They also preprocessed the images, using data augmentation (scaling and rotation of the images) and normalization (reducing the illumination and pose of the images). They achieved an accuracy of 89%, 75%, and 84% for happy, sad, and neutral expressions, respectively.

Regarding the division into positive, negative, and other classes, there are also works that have already tackled this. Li et al. [50] developed a framework called the Emotional Education Mechanism (EEM), inspired by the cognitive mode of the human being. This mechanism consists of a Knowledgeable Teacher Network (KTN), which consists of two ResNet50 networks, and a Self-Taught Student Network (STSN), which consists of two ResNet18 networks. There were three datasets used: RAF-DB, AffectNet, and FERPlus. Originally, RAF-DB and AffectNet datasets contain images from 7 different categories (the same categories present in the FER-2013 dataset), while FERPlus has one additional class (contempt). The authors divided the original labels into four categories: positive, negative, surprise and neutral. They also proposed a new supervised adaptive goal, called AdaReg loss, which helps to deal with class imbalance and increase the discriminatory

power of expression representation. It achieved an accuracy of 88.07% on RAF-DB, 63.97% on AffectNet and 90.49% on FERPlus.

Similarly to Li et al. [50], Uzun et al. [51] proposed a method for recognizing micro-expressions and classifying them into positive, negative, or surprising, without using the neutral group. The method is divided into four stages: preprocessing, feature extraction, feature selection, and classification. The initial process involved normalizing, aligning, and cropping the images. The FarneBack optical flow method was used to extract features from images [52]. Due to the sampling imbalance in the datasets, data augmentation techniques were utilized (rotating the samples 90°, 180°, and 270°). With the dataset balanced, via augmentation, they trained 5 distinct CNN models: VGG16, AlexNet [53], SqueezeNet [54], MobileNetV2 [55] and EfficientNetB0 [55]. The Particle Swarm Optimization (PSO) algorithm [52] filtered the best features of the images, resulting in the completion of the feature selection step. In the last stage, different kernels of the SVM algorithm [56] were used in the last stage. In this mechanism, they combined 3 datasets: SAMM, CASME-II and SMIC. The proposed framework achieved an accuracy of 87.84%.

FER using seven, five, and also three classes was done by Liu et al. [57], in which they constructed a model to identify micro-expressions called SQU-C3D, that combines SqueezeNet and C3D [58] methods. The proposed framework consists of three main stages: image preprocessing, apex frame identification, and micro-expression recognition. In the initial phase, the authors employed a Multitask Cascade Convolutional Neural Network (MTCNN) network [59] to capture and align the frames of the micro-expressions and locate 68 facial landmarks. In the next step, SqueezeNet was used to identify the apex frame. The final step involved feeding the C3D network the onset, apex, and offset frames for micro-expression recognition. The experiments were carried out using the same datasets as in the work of Uzun et al. [51]. The proposed framework achieved an accuracy of 80.29% with 7 classes (CASME II database), 81.33% with 5 classes (SAMM database), and 79.12% with 3 classes (SMIC-HS database, with positive, negative, and surprise), demonstrating that reducing the number of classes does not strictly lead to higher accuracy.

The division proposed in this paper follows the same division as in Canedo et al. [30]. They presented an algorithm for mood estimation, that utilizes facial expression recognition and pose estimation. For the facial expression recognition task, the authors proposed a CNN, which was trained using the CK+ dataset [60]. This dataset is composed of images from 7 categories, that have been subdivided into 3 new categories: negative, neutral, and positive. The images of the dataset were preprocessed, applying some techniques, such as rotation correction, cropping, and intensity normalization. The algorithm achieved an accuracy of 93% for the FER task.

2.3.2 Occlusion

To understand its causes, impacts, and potential implications in various domains, it is important to understand the intricacies of partial occlusion. This subsection focuses on questions aimed at understanding the methodologies used in facial expression classification, the techniques used to induce partial occlusion in facial structures, the algorithms utilized for expression classification, and the datasets researchers use. For a more comprehensive analysis, Tables 2.1, 2.2, and 2.3 illustrate the occurrence of upper occlusions, lower occlusions and other types of occlusion documented in the studies, respectively. These tables include the title of the paper, the type of occlusion with a visual representation, the method used to simulate partial occlusion (if applicable), and the exploration of the authors of several methods. The proposed method, which demonstrates superior performance, is highlighted in bold, along with the respective accuracy.

Table 2.1: Upper Occlusions

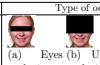
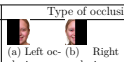
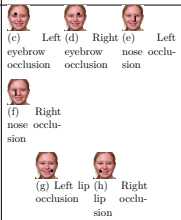
Paper	Type of occlusion	Simulation of partial occlusion	Methods	Datasets	Accuracy
Cheng et.al [61]	 (a) Eyes occlusion (b) Upper occlusion	Not-described	• Gabor • Local Gabor Binary Pattern Histogram Sequence (LGBPHS) • Multi-Scale Local Gabor Binary Pattern Histogram Sequence (M-LGBPHS) • **Gabor + Deep nonlinear network with 3 layers (proposed method)**	JAFFE	• Figure 2.1a: 82.86% • Figure 2.1b: 77.14%
Li et al. [62]	 (c) Eyes occlusion (d) Upper occlusion	Not-described	• Gabor • Fused Local Gabor Binary Pattern Histogram Sequence (F-LGBPHS) • **Gabor Filter + GLCM + K-Nearest Neighbors (KNN) + 10-fold cross-validation (proposed method)**	• JAFFE • RAF-DB	Figure 2.1c • 86.97% • 84.55% Figure 2.1d • 84.12% • 81.12%
Houshmand et al. [8]		• Detection of the face through grayscale images generated with modified HOG and linear SVM; • Estimation of 68 facial landmarks using the approach described in [63]; • Application of the VR patch following the dimensions of a Samsung Gear VR headset.	• VGG-Face (from scratch) + 5-fold cross-validation • ResNet50 (from scratch) + 5-fold cross-validation • **VGG-Face (transfer learning) + 5-fold cross-validation (proposed method)** • ResNet50 (transfer learning) + 5-fold cross-validation • VGG-Face + 5-fold cross-validation	• FERPlus • AffectNet • RAF-DB	• 79.98% • 50.13% • 73.37%
Poux et al. [64]		Not-described	• Symmetric auto-encoder + optical flow + 10-fold cross-validation + MSE. • Symmetric auto-encoder + optical flow + 10-fold cross-validation + Wing. • **Symmetric auto-encoder + optical flow + 10-fold cross-validation + EndPoint (proposed method)**.	CK+	87.10%
Petrou et al. [31]		• Detection of five facial landmarks (two for the center of each eye, one for the center of the nose, and two for the right and left side of the mouth); • With the landmarks of the nose and the eyes, and the distances of the algorithm described in [65], a rectangle is drawn on top of each image; • Upper part of the face is hidden to simulate the inclusion of VR headsets.	• Mini-Xception (pre-trained) • Mini-Xception (pre-trained + unfreeze last layer) • **Mini-Xception (pre-trained + unfreeze all layers) (proposed method)** • Mini-Xception (trained from scratch)	The authors developed their own dataset by combining five online datasets (FER 2013, Jafar Hussain Human, Unsplash, Pexels and Pixabay).	69.00%
Liu et al. [66]		Addition of black masks with different positions of the expression region (eyes, the mouth, left side of the face and right side of the face)	• Gabor Histogram • LBP Histogram • **Gabor multi-orientation features fusion + LGBPHS (proposed method)**	JAFFE	85.53%
Huang et al. [67]		• To simulate occlusion, graphically generated eyeglasses, medical masks, and random region masks were superimposed on un-occluded facial expression sequences; • AAM locates the facial points in each frame, resulting in the same generation process for eye, mouth, and lower-face occlusions for the next frames; • The distance between frames at the top of the nose is defined for random occlusions: after determining the position in the first frame, the patch is placed after the computed distance is adjusted in the next frame.	• Spatio-Temporal Local Binary Pattern (STLBP) • EdgeMap • Facial Soft Biometrics Estimation (FSE) • Compact Face Descriptor (CFD) • Compact Face Descriptor with Occlusion Detection (CFD-OD) • **Compact Face Descriptor with Occlusion Detection and Weight Learning (CFD-OD-WL) (proposed method)**	CK+	93.00%
Buciu et al. [68]		Superimposed black rectangles around the eyes and mouth regions to partially occlude them.	• Gabor + Co-Saliency Model (CSM) • **Gabor + Matthews Correlation Coefficient (MCC)**	• JAFFE • CK	• 84.00% • 92.30%
Zhi et al. [69]		To simulate partial occlusion on the facial images, an eye mask, nose mask, and mouth mask were made.	• **Graph-Structured Nonnegative Matrix Factorization (GSNMF) (proposed method)** • Sparse Nonnegative Matrix Factorization (SNMF) • Laplacianfaces	CK	93.30%
Mushfieldt et al. [70]		Black patches were applied to the eyes, mouth, and left and right sides of the face, superimposed.	• Gabor • Discriminative Nonnegative Matrix Factorization (DNMF) • **Viola-Jones face detection algorithm + Gaussian Mixture Models (GMMs) + LBP + SVM (proposed method)**	CK	68.00%
Rodrigues et al. [65]		• Obtaining 5 facial expression landmarks (two for the center of each eye, one for the center of the nose, and two for the right and left side of the mouth), using MTCNN; • Using the landmarks of the nose and eyes and the distances of the algorithm created by the authors, a rectangle is drawn on top of each image. This methodology simulates the presence of VR goggles.	• ResNet18 + MTCNN • VGG19 + MTCNN • **Combined (ResNet18 + VGG19) + MTCNN (proposed method)**	FER 2013	64.90%

Table 2.2: Lower Occlusions

Paper	Type of occlusion	Simulation of partial occlusion	Methods	Datasets	Accuracy
Cheng et.al [61]	 (a) Mouth occlusion (b) Lower occlusion	Not-described	• Gabor • LGBPHS • M-LGBPHS • **Gabor + Deep nonlinear network with 3 layers (proposed method)**	JAFFE	• Figure 2.2a: 82.86% • Figure 2.2b: 82.86%
Li et al. [62]	 (c) Mouth occlusion (d) Lower occlusion	Not-described	• Gabor • F-LGBPHS • **Gabor Filter + GLCM + KNN + 10-fold cross-validation (proposed method)**	• JAFFE • RAF-DB	Figure 2.2c • 90.90% • 81.73% Figure 2.2d • 86.66% • 74.90%
Poux et al. [64]	 (e) Lower occlusion (f) Mouth occlusion	Not-described	• Symmetric auto-encoder + optical flow + 10-fold cross-validation + Mean Squared Error (MSE). • Symmetric auto-encoder + optical flow + 10-fold cross-validation + Wing. • **Symmetric auto-encoder + optical flow + 10-fold cross-validation + EndPoint** (proposed method).	CK+	• Figure 2.2e: 70.20% • Figure 2.2f: 80.10%
Yang et al. [71]		Automatic wearing of a face mask, which automatically adds face masks that are shaped according to the orientation.	• Visual Geometric Group (VGG) • MobileNet • Region Attention Network (RAN) • Attention Convolutional Neural Network (ACNN) • Occlusion-Aware Deep Network (OADN) • **Two-stage attention model that consists of a binary deep classifier and a face-mask-aware FER deep classifier (proposed method)**.	• Multi-Label Face in the Wild for Facial Expression Recognition (M-LFW-FER) • Multi-Label KDDI Facial Expression Recognition (M-KDDI-FER)	• 87.92% • 90.01%
Liu et al. [66]		Addition of black masks with different positions of the expression region (eyes, the mouth, left side of the face and right side of the face).	• Gabor Histogram • LBP Histogram • **Gabor multi-orientation features fusion + LGBPHS (proposed method)**	JAFFE	92.11%
Huang et al. [67]	 (g) Mouth occlusion (h) Mouth occlusion (mask simulation)	• To simulate occlusion, graphically generated eyeglasses, medical masks, and random region masks were superimposed on un-occluded facial expression sequences; • AAM locates the facial points in each frame, resulting in the same generation process for eye, mouth, and lower-face occlusions for the next frames; • The distance between frames at the top of the nose is defined for random occlusions: after determining the position in the first frame, the patch is placed after the computed distance is adjusted in the next frame.	• STLBP • EdgeMap • FSE • CFD • CFD-OD • **CFD-OD-WL (proposed method)**	CK+	• Figure 2.2g: 73.54% • Figure 2.2h: 79.08%
Buciu et al. [68]		Superimposed black rectangles around the eyes and mouth regions to occlude them partially.	• Gabor + CSM • **Gabor + MCC**	• JAFFE • CK	• 83.50% • 87.20%
Zhi et al. [69]		To simulate partial occlusion in the facial images, an eye mask, nose mask, and mouth mask were made.	• **GSNMF (proposed method)** • SNMF • Laplacianfaces	CK	91.40%
Mushfieldt et al. [70]		Black patches were applied to the eyes, mouth, and left and right sides of the face, superimposed.	• Gabor • DNMF • **Viola-Jones face detection algorithm + GMMs + LBP + SVM (proposed method)**	CK	45.00%

Table 2.3: Other Occlusions

Paper	Type of occlusion	Simulation of partial occlusion	Methods	Datasets	Accuracy
Li et al. [62]	 (a) Left occlusion (b) Right occlusion	Not-described	• Gabor • F-LGBPHS • **Gabor Filter + GLCM + KNN + 10-fold cross-validation (proposed method)**	• JAFFE • RAF-DB	Figure 2.3a: • 89.69% • 87.73% Figure 2.3b: • 89.45% • 81.83%
M.D and Rahiman [72]	 (c) Left eyebrow occlusion (d) Right eyebrow occlusion (e) Left nose occlusion (f) Right nose occlusion (g) Left lip occlusion (h) Right lip occlusion	The occlusion was applied using MatLab code.	Combination of LBP and Symbolic Aggregate approXimation (SAX) feature extraction + Ensemble bag classifier (with supervised learning) (proposed method).	Fused database (JAFFE and YALE)	• Figure 2.3c: 88.81% • Figure 2.3d: 93.47% • Figure 2.3e: 93.25% • Figure 2.3f: 88.81% • Figure 2.3g: 90.93% • Figure 2.3h: 93.25%
Zhi et al. [69]		To simulate partial occlusion in the facial images, an eye mask, nose mask, and mouth mask were made.	• **GSNMF (proposed method)** • SNMF • Laplacianfaces	CK	94.00%
Mushfieldt et al. [70]	 (i) Right occlusion (j) Left occlusion	Black patches were applied to the eyes, mouth, and left and right sides of the face, superimposed.	• Gabor • DNMF • **Viola-Jones face detection algorithm + GMMs + LBP + SVM (proposed method)**	CK	• Figure 2.3i: 73.00% • Figure 2.3i: 56.00%

2.4 Summary

FER stands as a pivotal domain in deep learning, machine learning, and artificial intelligence, investigated extensively due to its broad array of applications. FER is integral to human-computer interaction, healthcare, security, and more. Nevertheless, the presence of occlusions—stemming from masks (as observed during the COVID-19 pandemic), glasses, scarves, and other facial coverings—continues to pose challenges in numerous real-world settings. Moreover, the significance of simplifying class divisions, such as grouping expressions into positive, negative, and neutral categories, is paramount. This streamlined classification facilitates more efficient and practical applications, while still capturing fundamental emotional variations.

Chapter 3

Methodology

This chapter details the methodology used to investigate the impact of occlusion on FER. The study used three CNN architectures: VGG19, ResNet18, and EfficientNetB1, trained on the FERPlus and FERV39K. The MaskTheFace algorithm was used to simulate surgical masks. An algorithm was also developed in order to simulate the presence of VR goggles. Furthermore, to assess the effect of class granularity, a reduced class set was applied to both datasets, consolidating similar expressions into three categories: negative, positive, and neutral.

3.1 Data Collection and Preprocessing

This section examines the stages of gathering and preprocessing the data, focusing on the FERPlus and FERV39K datasets. The section covers how the datasets were used, including the grouping of labels for consistent categorization of facial expressions. Additionally, it outlines the preprocessing techniques applied to simulate occlusions, such as goggles and mask occlusions, to enhance the datasets for more robust model training and evaluation. The section also details the process of categorizing the labels into three classes: positive, negative, and neutral.

3.1.1 Dataset Description

This subsection presents a detailed overview of the FERPlus and FERV39K datasets, both widely used in facial expression recognition research.

FERPlus dataset (Figure 3.1) is an extended version of the FER-2013 dataset [46], where the images were re-labeled into 8 classes of emotions: neutral, happiness, surprise, sadness, anger, disgust, fear, and contempt. It contains 35,887 grayscale 48*48 images: 28,709 for training and 3,589 for validation and testing processes. The dataset was created with the goal of correcting the FER-2013 dataset, specifically by fixing incorrectly classified images and removing images that did not have any faces. Each image was annotated in a CSV file with 12 columns: "Usage" (Training, PublicTest, and PrivateTest), "Image name", the 8 emotion labels, "unknown", and "NF" (not a face). This dataset was labeled by 10 crowd-sourced taggers, meaning that each person independently voted for the emotion they perceived in the image. Taggers chose one of eight emotions for every input image. By labeling every image multiple times, the process enabled emotions to be distributed rather than relying on a single annotation. By capturing the variability in human perception, this approach enabled a more robust labeling process [73].

Figure 3.1: Sample of the FERPlus dataset.

The FERV39K large-scale multiscene collection (Figure 3.2) marks a significant step forward in dynamic facial expression recognition, as it is specifically designed to enhance FER tasks. There are 38,935 video clips that are labeled with seven classic expressions - angry, disgust, fear, happy, neutral, sad and surprise, covering 22 intricate scenes (action, argue, business, conflict, contest, crime, crisis, daily life, elegant art, experiment, history,

interview, live show, medicine, official event, scholar report, school, social, speech, talk show, terror and war) in four different scenarios: everyday life, weak-interactive shows, strong-interactive activities, and anomaly problems [74].

Figure 3.2: Sample of the FERV39K dataset.

3.1.2 Data Preprocessing

In this part, the datasets were reorganized to fit the aims of the study. Preprocessing included simulating VR goggles through an occlusion algorithm and employing the Mask-TheFace algorithm to mask occlusions. Furthermore, the method of categorizing the datasets into three separate labels is detailed, maintaining uniformity in the analysis.

In order to reclassify the samples in FERPlus dataset, the columns “unknown” and “NF” were removed because they are not necessary to perform any classification in this investigation. Then, it was necessary to calculate the highest probability value per line (that is, the 8 emotions are numbered from 0 to 10, with 0 being the lowest probability and, consequently, 10 being the highest probability, preferring a total of 10 values per line). In many cases, the highest probability value was repeated on the same line. The manual check of all images to determine which of the emotions would be most likely, would become a very painful and time-consuming process, so the choice, in this case, was made randomly. After this process, a new CSV file was created, with 2 columns (“Image name” and “label”), with the new labels (0=neutral, 1=happy, 2=surprise, 3=sad, 4=anger, 5=disgust, 6=fear, and 7=contempt).

FERV39K consists of 38,935 video clips organized by daily scenes. For this study,

the videos were consolidated into a single folder, as the aim was not to compare FER across different daily scenes. All frames from the video clips were then extracted, totaling 1,129,744 images, which were combined and divided into training (80%), testing (10%), and validation (10%) sets. Similarly to FERPlus, the dataset was re-labeled with the following categories: 0 for neutral, 1 for happy, 2 for surprise, 3 for sad, 4 for anger, 5 for disgust, and 6 for fear.

Due to the absence of images with occlusion in the datasets, it was necessary to pre-process the images to introduce and simulate occlusions for more comprehensive analysis. In order to simulate the presence of VR goggles (Figure 3.3), an occlusion algorithm was developed[1], as described in the previous work developed [65]. The algorithm is divided into 2 fundamental steps: obtaining the location of the face and the facial landmarks using an MTCNN [59] and then calculate the position of the goggles [65] (1).

Figure 3.3: Sample of the FERPlus dataset with occlusion.

More specifically, firstly, the MTCNN network was used, which outputs the positions of 5 facial landmarks (eyes, mouth - left and right corners, and nose), which consists of three stages:

1. Proposal Network (P-Net) - convolutional network without dense layers that aims to obtain candidate facial windows and the corresponding bounding box regression vectors (coordinates of the vertex of the square where the detected face encloses). After this process, highly overlapped candidates are merged using a Non-Maximum Suppression (NMS);

[1]https://github.com/SofiaRodrigues41737/Occlusion_Algorithm

2. Refine Network (R-Net) - receives the results of P-Net, and rejects the false candidates, performing calibration with bounding box regression and conducting NMS. This network has 3 outputs: face classification, bounding box regression, and facial landmark location;

3. Output Network (O-Net) - similar to R-Net, receives the output of R-Net and identifies face regions with more supervision. The O-Net output displays the positions of 5 facial landmarks (eyes, mouth, and nose).

Using the facial landmarks obtained, the middle point between the eyes and the distance between them was calculated. Secondly, it estimates the width and height of VR glasses as 20% greater than the distance between the eyes and 150% the distance between the eye contour and the nose. By calculating the tilt angle, the rectangle on top of the sample image is drawn in gray. In Figure 3.3 it is possible to verify that the algorithm is very robust, simulating occlusion in almost all samples, from images with people who are wearing glasses to images of people with different face poses (lateral or frontal face). In addition, it is possible to verify that FERPlus has some images with low quality, where it is not possible to perceive the facial landmarks, therefore, the algorithm did not simulate the occlusion in these samples. When this occurs, the affected images are excluded from the dataset, ensuring that only images with actual occlusion are retained.

To simulate mask occlusion, a masking algorithm was additionally applied as detailed in [75]. The algorithm is composed of MaskTheFace (Figure 3.4), a computer vision-based script designed to apply masks to faces in images. Using the dlib-based face landmarks detector, the algorithm identifies the tilt of the face and six critical facial features essential for accurate mask application. Depending on the detected face tilt, an appropriate mask template is selected from a pre-defined library of masks. The template is then precisely transformed according to the six key facial features to ensure a perfect fit on the face. The accuracy of MaskTheFace lies in its ability to identify all faces in an image and apply masks to them that have been selected by the user. To ensure realistic and accurate mask placement, the algorithm takes into account various factors like face angle, mask

Algorithm 1 Occlusion Algorithm

procedure MAKEGOGGLES(sample, landmarks)
  $left\_eye\_x, left\_eye\_y \leftarrow landmarks[0][0], landmarks[0][5]$
  $right\_eye\_x, right\_eye\_y \leftarrow landmarks[0][1], landmarks[0][6]$
  $nose\_x, nose\_y \leftarrow landmarks[0][2], landmarks[0][7]$
  $middle\_x, middle\_y \leftarrow \frac{right\_eye\_x+left\_eye\_x}{2}, \frac{right\_eye\_y+left\_eye\_y}{2}$
  $googles\_width = 2.2*\sqrt{(right\_eye\_y - left\_eye\_y)^2 + (right\_eye\_x - left\_eye\_x)^2}$
  $googles\_height = 1.5 * \sqrt{(middle\_eyes\_y - nose\_y)^2 + (middle\_eyes\_x - nose\_x)^2}$
  $rectangle = (0, 0, googles\_width, googles\_height)$
  $middle\_rectangle\_x, middle\_rectangle\_y = \frac{googles\_width}{2}, \frac{googles\_height}{2}$
  $angle = \frac{right\_eye\_y-left\_eye\_y}{right\_eye\_x-left\_eye\_x} * \frac{180}{\pi}$
  $rectangle = rectangle.rotate(-angle, (middle\_rectangle\_x, middle\_rectangle\_y))$
  $final\_size = rectangle.size$
  $sample.paste(rectangle, \frac{middle\_eyes\_x-final\_size[0]}{2}, \frac{middle\_eyes\_y-final\_size[1]}{2})$
end procedure

fit, and lighting conditions. By processing a single image or an entire directory of images, it can be used as an invaluable tool for converting existing face datasets into masked-face datasets. The algorithm offers several mask types, including 'N95', 'surgical_blue', 'surgical_green', 'cloth', 'empty', and 'inpaint'. It is also possible to choose their preferred mask type and enhance it with various patterns and colors. Since the study intends to examine the impact of partial occlusion in FER, only the conventional surgical mask was used (Figure 3.5).

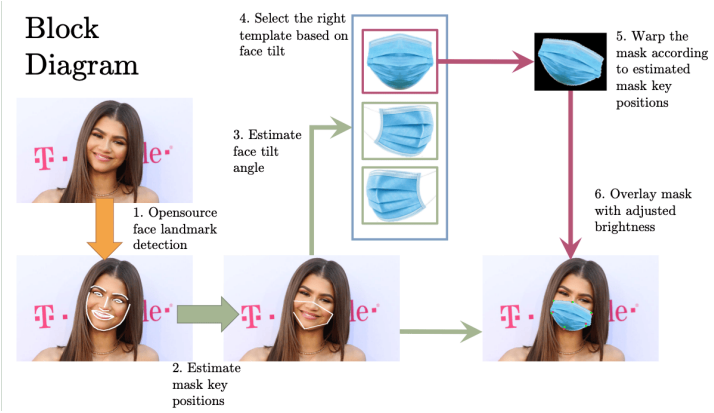


Figure 3.4: Schematic representation of MaskTheFace algorithm [75].

When performing FER, similar expressions can be grouped, as in some applications, the indication of a positive, negative, or neutral expression is sufficient [30]. This type

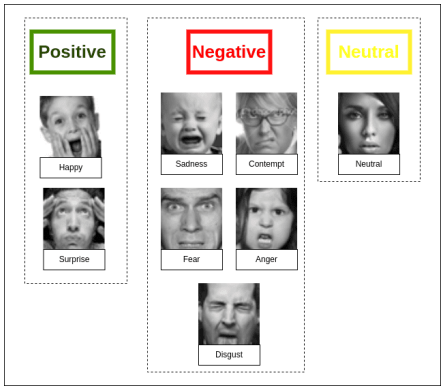
Figure 3.5: Sample of the FERPlus dataset with mask occlusion

of simplification can help reduce the complexity of the classification problem and may improve the generalization of the model. However, it should be noted that some level of detail is lost in the data, which can be advantageous (simpler classification) or harmful (loss of specific emotional information). When grouping all data into fewer classes, it is also critical to ensure that the data are balanced in some way.

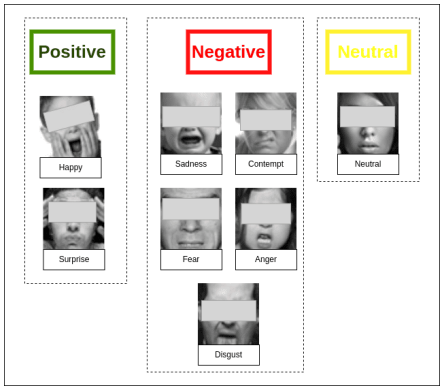
At this stage, the images were reclassified from 8 classes (FERPlus) and 7 classes (FERV39K) into 3 categories. To categorize the new labels, emotions such as fear, disgust, sadness, anger, and contempt were grouped as class 0 (negative expressions) for the FERPlus dataset. In the FERV39K dataset, fear, disgust, sadness, and anger were grouped as class 0 (negative expressions). Happy and surprise were classified as class 1 (positive expressions), while neutral was assigned to class 2 (Figure 3.6).

3.2 CNN Architectures

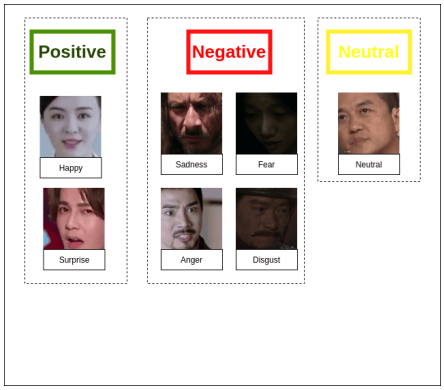
CNNs have transformed image recognition and become a key component of modern CV. Models are empowered by CNNs to extract and learn complex hierarchical features from visual data, including low-level edges and high-level semantic concepts. Tasks such as image classification, object detection, and facial recognition have been revolutionized by this capability, leading to unprecedented accuracy and efficiency. CNNs learn features directly from data during training, which is different from traditional machine learning methods which rely on manual feature engineering. This adaptability makes this architectures particularly well-suited for handling the diverse variability present in real-world



(a) FERPlus



(b) FERPlus with eyes occlusion



(c) FERV39K

(d) FERV39K with eyes occlusion

Figure 3.6: Class grouping - Positive, Negative and Neutral

visual datasets [76].

3.2.1 Overview of CNNs for Image Recognition

CNNs represent a specialized subset of DL models tailored for processing grid-like data structures, such as images. In contrast to conventional neural networks that handle input features independently, CNNs exploit the spatial configuration of data, allowing for more efficient capture of pixel interrelations and feature extraction. This has revolutionized CV by eliminating the need for human experts to develop image recognition features manually. Instead, the networks autonomously uncover these features through a hierarchical learning process, initially identifying simple elements like edges and textures, and subsequently recognizing more complex patterns such as shapes and objects in later layers [77].

The foundation of CNNs comprises several crucial layers that function together to analyze and categorize images (e.g. Figure 3.7). Convolutional layers (Conv) use a range of filters (also known as kernels) on the input image, convolving over the pixels to identify local patterns like edges or textures and gradually more complex features like shapes. The network trains to adjust the weights of these filters, enabling it to concentrate on the most pertinent elements [77].Weights play a crucial role in how neural networks learn. They determine the strength of connections between neurons across layers. In a CNN, every filter has its own set of weights, which are used on the input data via convolution operations. This capability allows CNNs to recognize spatial hierarchies within the data, where lower layers detect basic features, such as edges or corners and deeper layers uncover more abstract concepts like shapes [78].Following convolution, feature maps are generated. These are the results of applying filters to the original data. Each feature map emphasizes various components of the input, such as edges, textures, or patterns, based on the detected attributes by the filters. To enhance efficiency, pooling layers (Pool) are implemented. These layers conduct downsampling, a technique that shrinks the feature maps by summarizing areas within them (such as taking either the maximum or average of small sections). By downsampling, the network preserves crucial features while

eliminating redundant details, thereby decreasing the data volume for processing and accelerating computation. Pooling also enhances the robustness of the model against minor input variations, like shifts or rotations, thereby improving its generalization capacity. In the final stages, fully connected layers use the high-level features derived from earlier layers to make predictions, generally classifying the input into particular categories. These layers establish connections between each neuron in one layer and every neuron in the subsequent layer, merging the acquired feature information to generate the resultant output, such as determining the class of the object in the image.

In CNNs, activation functions serve as an essential component, as they introduce non-linearity after each convolutional and fully connected layer. Key to the ability of the network to learn and represent the intricate, non-linear links found in real-world datasets, activation functions like Softmax are indispensable. This non-linear behavior is fundamental for tasks like image recognition, where the connection between pixels and higher-level features like edges, textures, or objects is inherently non-linear. Without these functions, CNNs could not establish the complex decision boundaries necessary for distinguishing between images.

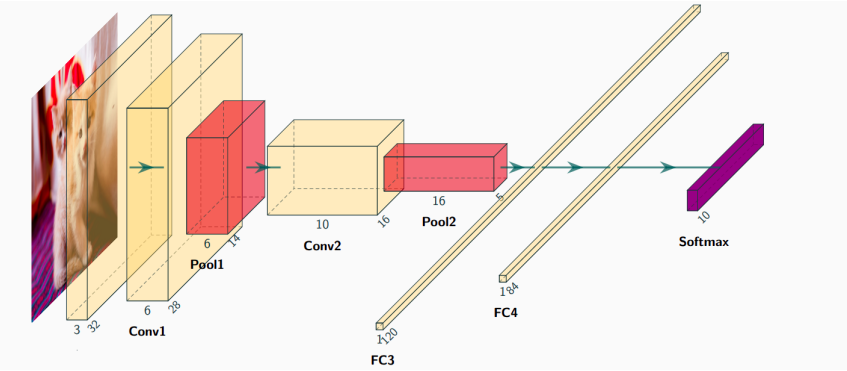


Figure 3.7: CNN Example: LeNet-5 [79].

The capacity of CNNs for hierarchical learning allows them to represent images at multiple abstraction levels. In the initial layers, they can identify elementary features such as edges and gradients, while in deeper layers, they discern complex patterns and complete objects or scenes. This capability, called translation invariance, is crucial for

object detection and scene classification in intricate settings, as it enables the recognition of objects regardless of their placement in the image [77].

3.2.2 Selected CNN Architectures

This subsection thoroughly explores CNN architectures, focusing on VGG19, ResNet18, and EfficientNetB1 as three prevalent models. The unique structures and design principles of these architectures are described and underscored, illustrating their efficacy in diverse image recognition applications.

Visual Geometric Group 19 (VGG19) (Figure 3.8) is a variant of VGG architectures, with 19 deeply connected layers [80]. In this work, a pre-trained model was used, trained on the ImageNet dataset of 1.3 million images, consisting of 1000 classes (proved to be more robust than without any prior knowledge in previous work [65]). VGG19 is composed of fully convolutional and fully connected layers that are highly connected, resulting in better feature extraction. The kernel size is 3*3 and the input size is 224*224*3. This model has a structure that allows better extraction of image features, using Maxpooling for downsampling (applied to improve the anti-distortion ability of the network to the image) and using a Rectified Linear Unit (ReLU) as the activation function.

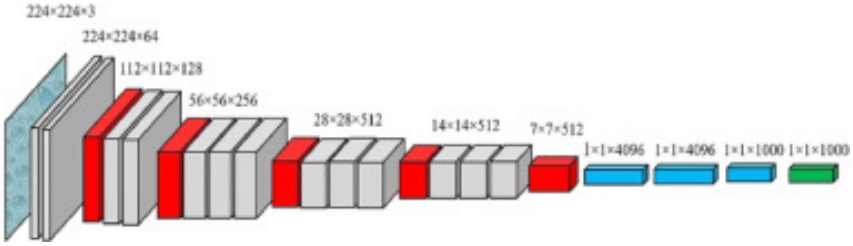


Figure 3.8: Architecture of VGG19 [80].

Residual Network 18 (ResNet18) (Figure 3.9) [81], was also pretrained on the ImageNet dataset and consists of 18 layers (17 convolutional layers, a fully connected layer and an additional softmax layer to perform the classification task). The convolutional layers use 3*3 filters and the input size is 224*224*3. Downsampling was also used, which is performed by convolutional layers with a stride of 2. There is an average pooling followed

by a fully connected layer with a softmax layer. In this network, residual connections are inserted between layers.

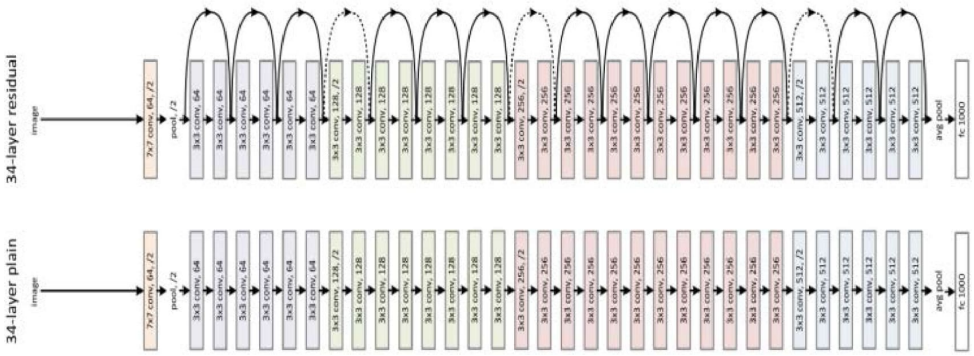


Figure 3.9: Architecture of ResNet18 [80].

EfficientNetB1 (Figure 3.10) is composed of a basic component, a Mobile Inverted Bottleneck Convolution (MBConv) module. In this component, the channels of the features are changed, using a 1*1 convolution that is followed by a depth-wise convolution. Then, a channel attention mechanism is introduced (the Squeeze-and-Excitation Network (SENet) mechanism) [82]. Lastly, the channels of the feature maps are reduced using a 1*1 convolution.

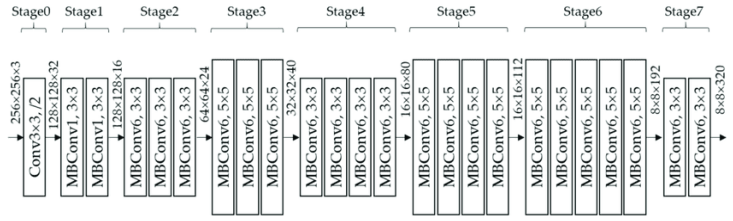


Figure 3.10: Architecture of EfficientNetB1 [82].

3.2.3 Model Hyperparameters

To improve the training process of the model, a variety of hyperparameters were employed. These included the learning rate, batch size, optimizer, number of epochs, as well as regularization strategies and techniques for computing loss.

The learning rate is a crucial hyperparameter that determines the size of the steps taken towards minimizing the loss function during the training process, controlling how much to adjust the weights of the model in response to the calculated gradients [77]. In the models used in this work, the learning rate was set to 0.001. A smaller learning rate could lead to more stable convergence but could require more epochs to achieve an optimal solution. Conversely, a larger learning rate could accelerate training but risked overshooting the optimal weights, potentially resulting in divergence.

Batch size describes the quantity of training samples handled before updating the internal parameters of the models, essentially determining how many examples are processed in a single iteration of model training [77]. Here, a batch size of 64 achieved a compromise between memory efficiency and training speed. Smaller batch sizes tended to regularize and improve generalization, whereas larger batch sizes decreased the duration of each epoch.

The optimizer is an algorithm designed to minimize the loss function by adjusting the parameters of the models based on the computed gradients. In this instance, the Stochastic Gradient Descent (SGD) optimizer was utilized (Eq. 3.1). This optimizer was chosen for its simplicity and effectiveness [83]. The momentum parameter was set to 0.9, which helped accelerate the optimization process in significant directions, smoothing the parameter updates and reducing oscillations.Momentum stabilizes weight updates by considering previous gradients rather than relying solely on the current gradient. A momentum value of 0.9 indicates that the optimizer incorporates past weight updates to enhance the consistency of the adjustments. Additionally, weight decay, set at 5^{-4}, acted as a regularization method to combat overfitting; it penalized larger weights during the optimization process, promoting simpler weight configurations to improve the generalization capabilities of the models.

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i)}, y^{(i)}) \tag{3.1}$$

, where θ symbolizes the vector of model parameters that are updated during training,

while η represents the learning rate, a hyperparameter that determines the magnitude of steps taken in each iteration of the optimization process. The expression $\nabla_\theta J(\theta; x^{(i)}, y^{(i)})$ stands for the gradient of the cost function J with respect to the parameters θ, calculated using the i-th pair of training data $(x^{(i)}, y^{(i)})$. Furthermore, $(x^{(i)}, y^{(i)})$ describes the i-th training instance, where $x^{(i)}$ denotes an input (like an image), and $y^{(i)}$ represents the associated label (such as the class of the image).

An epoch refers to one complete cycle through the training dataset. VGG19 and ResNet18 were trained for 100 epochs, whereas EfficientNetB1 utilized 200 epochs. This allows the model to iteratively learn from the dataset. However, monitoring the performance of the models on a validation set is crucial to avoid overfitting. Excessively training without adequate regularization might lead to a model that performs well on the training set but fails to generalize to unfamiliar data.

The main regularization approach used in this model is weight decay, which is integrated into the settings of the optimizer. This technique mitigates the overfitting of the models by applying a penalty on large weights, thus promoting generalization and enhancing performance on new data.

To tackle the imbalance observed in the FERPlus and FERV39K datasets, a weighted loss function was applied. This involved incorporating a weight parameter into the CrossEntropyLoss function, as described in Eq. 3.2. The purpose of the loss function is to determine how accurately the predictions of the models align with the actual targets. Tailored for multiclass classification, it compares the predicted probabilities with the true class labels. The inclusion of a weight parameter allows the model to assign different importance levels to each class, which is particularly beneficial for imbalanced datasets [84]. By calculating these weights based on the training datasets, the models can focus more on underrepresented classes, thus enhancing their learning process.

$$loss(x, y) = -weight[y] * \frac{log(exp(x[y]))}{sum(exp(x))} \tag{3.2}$$

, where x represents the model output, y denotes the target class, exp signifies the

exponential function, and $sum(exp(x))$ indicates the sum of exponentials in all classes.

3.3 Training Process

This section provides a detailed overview of the experimental framework, encompassing the hardware and software employed, the training protocol adopted—emphasizing the supervised learning strategy along with the validation and testing strategies utilized, including the performance evaluation metrics, and the use of transfer learning within the chosen CNN architectures.

3.3.1 Experimental Setup

The training of the networks was executed on a machine equipped with a 64GB RAM AMD Ryzen Threadripper 3970X 32-Core Processor alongside an NVIDIA GeForce RTX 3090 Graphics Processing Unit (GPU). The software environment included the X2Go Client, a remote desktop solution that allows users to access and control a remote machine via a graphical interface over a network connection. This client was employed to connect via Virtual Private Network (VPN) to Escola Superior de Tecnologia e Gestão (ESTIG), operating on Ubuntu with the MATE desktop environment.

3.3.2 Training Protocol

The models were trained using a supervised learning approach, a type of ML that entails training a model with labeled data. In this setup, each instance includes an input-output pair, specifically images as inputs and their corresponding class labels as outputs. The aim of supervised learning is to discover a function that precisely maps inputs to outputs by minimizing the error between predicted outputs and actual labels. This process equips the models to make correct predictions on unseen data [85]. In this setting, backpropagation played a crucial role as a fundamental algorithm in neural network training, enabling the model to learn from the labeled pairs to enhance prediction accuracy.

The training process comprised several stages enabling the model to effectively learn from the dataset. Initially, the input data was introduced to the neural network, prompting predictions based on the current weight assignments of the models. After this forward pass, the loss function quantified the deviation between the predicted outputs and the actual labels, evaluating the prediction accuracy of the models. Once the loss was assessed, a backward pass ensued, where backpropagation computed the gradients of the loss relative to each weight. This procedure entailed propagating the error in reverse through the network to assess how each weight impacted the overall loss. Using these gradients, the optimizer adjusted the weights to reduce loss, thus refining the parameters of the models to enhance accuracy. This weight adjustment cycle was repeated over multiple epochs, incrementally adjusting the model until it reached a satisfactory level of performance (Figure 3.11).

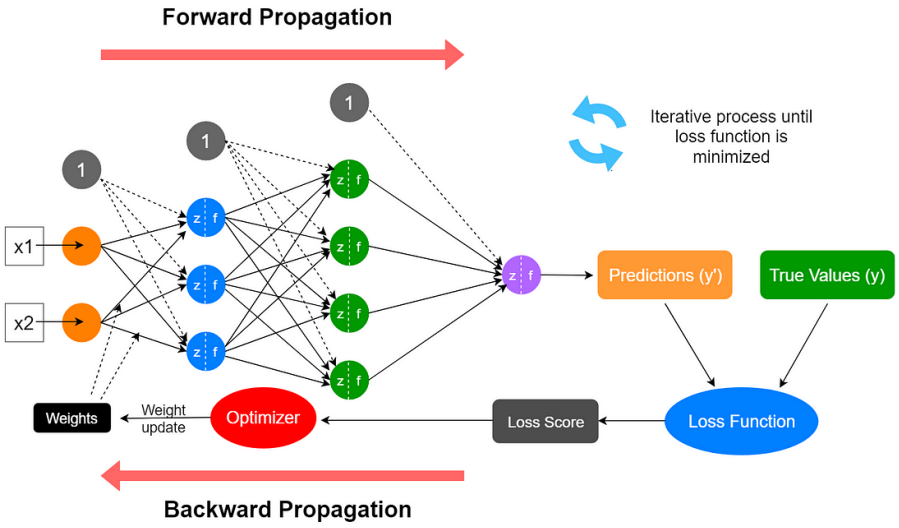


Figure 3.11: Backpropagation process [86]

Avoiding overfitting is crucial during the training process to ensure that the models generalize well to unseen data, rather than simply memorizing the training examples [87]. In this study, two key methods were employed, in order to prevent overffiting:

- Weight Decay: To address overfitting, this regularization method was employed by

imposing a penalty on larger weights during training. By incorporating a penalty in the loss function associated with the weight magnitude, weight decay promotes less complex models, thereby reducing the tendency of the network to overfit the training data. This strategy was effective in mitigating overfitting, eliminating the need for early stopping;

- Saving the Best Model: During the training phase, the model exhibiting the best performance on the validation set was preserved. This tactic prevented any potential deterioration in performance due to extended training, enabling the models to undergo complete training without the threat of overfitting. The best model could be reloaded when needed, thus ensuring optimal generalization;

To effectively adjust the numerous parameters of the intricate architectures applied, EfficientNetB1, VGG19, and ResNet18, a prolonged training period is required. The best-performing model was retained while weight decay was utilized to prevent overfitting while allowing these models to fully converge. Weight decay acts as a regularization technique to prevent large weights, enhancing generalization. While preserving the best-performing model, it ensured optimal validation performance. By using these strategies, training was extended while maintaining robustness, leading to optimal performance in complex neural network architectures.

3.3.3 Validation and Testing

To assess models performance, two metrics were consistently used during both validation and testing stages. Cross-Entropy Loss served as the main metric for quantifying the difference between predicted class probabilities and actual labels, offering crucial insights for model optimization. Additionally, accuracy was calculated to gauge the percentage of correctly predicted labels, acting as a broad measure of the effectiveness of the models.

3.3.4 Transfer Learning

In many DL tasks, especially when working with limited datasets, training a model from scratch can be inefficient. To address this, transfer learning was employed, allowing models to leverage previously learned knowledge. Transfer learning is an ML strategy in which a model created for a specific task is utilized as the foundation for a model on another, related task. By employing transfer learning, a model can take advantage of knowledge acquired from addressing a prior issue, instead of building a model from the ground up. This approach decreases the data, time, and computational resources required for training. Within DL, transfer learning generally entails employing pre-trained models—those trained on extensive and versatile datasets such as ImageNet — and modifying them for a new task. This involves immobilizing some of the early layers, which capture common characteristics like edges and textures (frozen layers), and only training or adjusting the final layers that are tailored to the specific task [88] (Figure 3.12).

In the VGG19 model, the convolutional layers were kept unchanged while adjustments were made and training conducted on the newly appended fully connected layers. The last layer of the classifier was substituted to correspond with the class count of the datasets. Similarly, in ResNet18, the preceding residual blocks were left unaltered, with only the terminal fully connected layer being replaced and fine-tuned. For EfficientNetB1, most of the initial layers responsible for extracting general features remained unchanged, while the final classification layer was replaced and optimized for the specific dataset.

3.4 Evaluation Metrics

To evaluate the performance of the trained models, several evaluation metrics were applied. These metrics offer a look at various dimensions of model accuracy and the effectiveness of classification. This section details the main metrics used for evaluation: accuracy, confusion matrix, precision, recall, F1 score, and per-class accuracy, providing a broad understanding of the performance of the models across different classes.

Accuracy is a crucial metric for assessing the performance of the models by determining

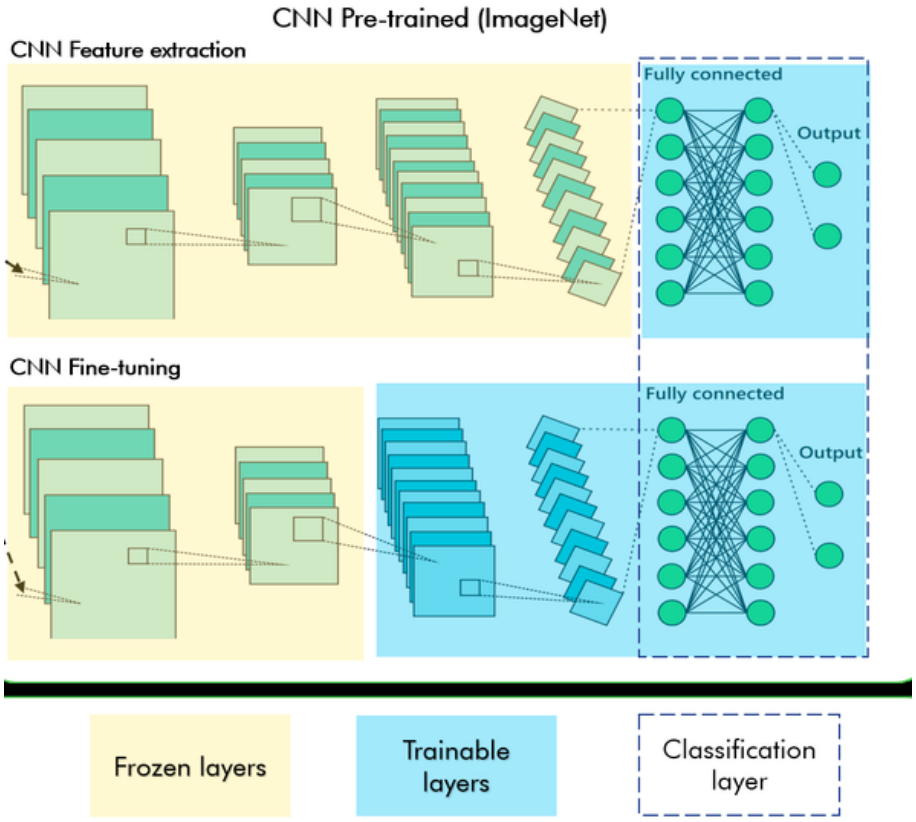


Figure 3.12: Scheme of transfer learning process [89]

the ratio of correct predictions to the total number of predictions. It indicates the capacity of the models to accurately identify both positive and negative instances, offering a general overview of its efficacy (Eq. 3.3).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.3}$$

Where:

- TP (True Positives): Cases where the model correctly predicted the positive class.
- FP (False Positives): Cases where the model incorrectly predicted the positive class.
- FN (False Negatives): Cases where the model incorrectly predicted the negative class.
- TN (True Negatives): Cases where the model correctly predicted the negative class.

The confusion matrix offers an in-depth understanding of a classification performance of the models by displaying the distribution of predictions among various classes:

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$$

Precision assesses the correctness of positive predictions by calculating the ratio of true positives to the total of positive predictions made. It shows how effectively the model minimizes false positives, thus indicating its dependability in forecasting the positive class (Eq. 3.4).

$$Precision = \frac{TP}{TP + FP} \tag{3.4}$$

Recall, sometimes referred to as Sensitivity or True Positive Rate, measures the capacity of the models to detect all pertinent positive samples within a dataset. This metric is determined by dividing the number of true positive predictions by the total actual positive

cases, thus offering an understanding of how well the model identifies positive instances while reducing false negatives (Eq. 3.5).

$$Recall = \frac{TP}{TP + FN} \tag{3.5}$$

The F1 score represents the harmonic mean of precision and recall, offering a singular measure that balances the compromise between these metrics. This score is notably valuable in situations where class distributions are imbalanced, as it underscores both the accuracy of positive predictions (precision) and the capacity of the models to identify all pertinent positive cases (recall). It is computed using Eq. 3.6 or Eq. 3.7.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{3.6}$$

Alternatively:

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{3.7}$$

Per-class accuracy evaluates the effectiveness of the models for each category separately, providing an in-depth assessment of its classification ability per class. It is determined by computing the accuracy for each class using the confusion matrix, which concisely compares the predictions of the models with the actual labels. For a particular class i, the per-class accuracy is given by Eq. 3.8.

$$\text{Per-class Accuracy}_i = \frac{TP_i}{TP_i + FN_i} \tag{3.8}$$

Where:

- TP_i is the true positives for class i,
- FN_i is the false negatives for class i.

3.5 Comparative Analysis of Architectures

This section provides a comparative analysis of three CNNs used in this work: VGG19, ResNet18, and EfficientNetB1. The focus of this analysis is on their performance in image recognition tasks, computational efficiency, as well as the challenges and limitations associated with each model.

3.5.1 Performance Comparison

Comparing models and architectures in ML, especially for CNNs, involves a structured evaluation approach. Initially, models are trained and tested on benchmark datasets like ImageNet to ensure an equitable comparison of performance metrics across various architectures. Essential performance indicators, such as accuracy, precision, recall, F1 score, and loss, are computed to evaluate the efficacy of each model, offering quantitative data that simplifies comparative analysis.

Cross-validation techniques, for instance, cross-validation itself, are used to verify that performance metrics are reliable and not excessively reliant on a specific train-test division. This approach facilitates a more accurate assessment of each ability of the models to generalize. Moreover, the duration required to train each model is documented, along with the computing resources expended (such as GPU memory and Central Processing Unit (CPU) usage), offering vital insights into the efficiency and feasibility of implementing each architecture in actual applications.

The speed at which inference occurs, quantifying the time for each model to generate predictions, is particularly critical for applications that demand real-time processing. Additionally, each resistance of the models to overfitting is assessed by analyzing the training and validation loss curves to gauge the ability of the model to generalize to new data. Typically, a model is favored if it sustains low validation loss while attaining high accuracy.

The total count of parameters in each architecture is also crucial consideration, affecting both the duration of training and the demand for memory; models with a reduced

number of parameters might operate more efficiently, especially in environments with limited resources. Additionally, qualitative analyses like confusion matrix visualizations offer valuable insights into the performance of the models beyond mere numerical metrics, aiding in the comprehension of specific strengths and weaknesses of each architecture.

3.5.2 Computational Efficiency

The application of CNN architectures is heavily shaped by their computational efficiency. The computational demands and performance aspects of models like VGG19, ResNet18, and EfficientNetB1 differ significantly. VGG19, which encompasses about 144 million parameters [80], required significant resources, leading to high memory consumption and slower processing speeds. This deep architecture demanded ample computational power for both training and inference. Conversely, ResNet18, with approximately 11 million parameters [81], presented a more efficient option. The introduction of residual connections in ResNet18 addressed the vanishing gradient issue (which occurs when gradients become exceedingly small as they are backpropagated through deep networks, leading to ineffective weight updates for earlier layers) and improved both inference speed and memory usage over VGG19. As a result, ResNet18 is ideal for scenarios where speed and resource efficiency are paramount, making it suitable for real-time image classification tasks. EfficientNetB1 introduced an strategy to balance computational efficiency with high performance. With around 7.8 million parameters [82], it reduced the computational load required for training and inference. Its compound scaling method optimizes model depth, width, and input resolution, ensuring swift inference and low memory utilization.

3.5.3 Challenges and Limitations

In the context of the challenges faced during model training, class imbalance emerged as a significant issue. Both the FERPlus and FERV39K datasets presented uneven distributions of examples across various emotion classes, which lead to biased model performance. When certain classes are underrepresented, the model struggle to learn to recognize those

emotions accurately, resulting in lower overall performance for those classes. The machine utilized for training, equipped with an AMD Ryzen Threadripper 3970X 32-Core Processor and an NVIDIA GeForce RTX 3090 GPU, was shared among multiple users. The common use of this environment frequently led to competition for resources, which posed challenges in obtaining the requisite computational capacity for model training. This occasionally rendered the GPU inaccessible, disrupting the training process. Consequently, the models needed to be trained repeatedly, complicating the overall schedule and potentially affecting the outcomes.

3.6 Summary

The processes of assembling the FERPlus and FERV39K datasets included reorganizing the data by eliminating extraneous columns and adopting a novel labeling system, prioritized by the highest probability values. A masking algorithm was used to replicate the effect of VR goggles on facial images, facilitating occlusion analysis. To improve the training process, various hyperparameters were used, such as the learning rate, batch size, optimizer, number of epochs, and regularization strategies like weight decay and saving the best model to mitigate overfitting. The training of models was conducted using a supervised learning method, leveraging backpropagation for efficient learning through forward and backward steps, loss calculation, and weight updates. To evaluate the performance of the models, metrics like accuracy, confusion matrix and per-class accuracy were applied, offering insights into classification performance. The computational demands of chosen CNN architectures—VGG19, ResNet18, and EfficientNetB1—differed notably, with VGG19 demanding considerable resources due to its extensive parameter size, whereas ResNet18 and EfficientNetB1 provided more efficient alternatives. ResNet18 managed the vanishing gradient problem, enhancing training reliability, and EfficientNetB1 balanced computational efficiency with performance via optimized scaling. Training challenges included dataset class imbalance leading to skewed performance and limited access to shared hardware resources, affecting GPU availability and requiring multiple training sessions,

potentially influencing the overall results.

Chapter 4

Results and Discussion

The results of the study on the impact of occlusion in mask and goggles scenarios, along with the implications of class reduction, will be discussed in this chapter.

4.1 Review of the Results

Regarding overall accuracy, using FERPlus dataset, in the no occlusion scenario, ResNet18 demonstrated the highest accuracy of 86.1%, followed by VGG19 with 83.0%, and EfficientNetB1 with 82.3% (Table 4.1a).

When simulating the use of VR goggles, the EfficientNetB1 performed best, achieving an accuracy of 79.7%, followed by VGG19 with 79.5%, and ResNet18 with 77.3% (Table 4.1b). With mask occlusion, EfficientNetB1 emerged as the top classifier with an accuracy of 71.2%, followed by VGG19 with 70.4% and ResNet18 with 69.5% (Table 4.1c).

In a more detailed analysis, in the no-occlusion scenario, ResNet18 exhibited the lowest recognition rates for contempt at 28.0% and fear at 36.6%, while achieving the highest recognition rates for happiness at 92.3% and neutral at 86.1%. In VGG19, the two main recognized emotions were neutral (86.1%) and happiness (92.3%), whereas the worst performances matched those of ResNet18, with contempt at 24.0% and fear at 40.0%. Similarly, in EfficientNetB1, the leading emotions were neutral (86.5%) and happiness (92.9%), while the least recognized emotions mirrored those of the other networks, with

contempt at 24.0% and fear at 38.3%.

When upper occlusion was introduced, the recognition rates for emotions with the lowest accuracy in ResNet18 remained consistent with the previous scenario, staying at 28.0% for contempt and 30.0% for fear. Similarly, in VGG19 and EfficientNetB1, this pattern persisted, with contempt at 20.0% and 24.0% respectively, and fear at 38.3% and 35.0% correspondingly. In contrast, emotions with the highest recognition rates remained consistent with the previous scenario, with ResNet-18 achieving rates of 89.9% and 82.3% in that order. Likewise, in VGG19, EfficientNetB1 the top two emotions remained unchanged from ResNet18, with happiness reaching rates of 91.1%, 92.3%,respectively, while neutral attained rates of 83.6% and 84.7% correspondingly.

Turning to the scenario involving mask occlusion, in ResNet18, the lowest emotions recognition rates persisted at 20.0% for contempt and 30.0% for fear. VGG19 and EfficientNetB1 are consistent with their earlier scenarios, where the lowest accuracy achieved is 20.0% being contempt, and 40.0% and 33.3% being fear, respectively. However, the highest recognition rates showed a change in ResNet18; while happiness maintained its position as one of the top two emotions with 75.2%, surprise emerged as the new leading emotion at 78.7%. This shift is also evident in VGG19, with rates of 80.1% for happiness, and 81.2% for surprise. In EfficientNetB1, the highest scores are for neutral and happiness, at 76.0% and 81.5% respectively.

The results from the FERPlus dataset indicate that emotions such as contempt and fear are particularly challenging to recognize across all scenarios, as evidenced by consistently low recognition rates that persist even under occlusion conditions. These low rates suggest that contempt and fear often involve subtle facial cues that are easily obscured, making accurate identification difficult. Conversely, emotions like happiness and neutral consistently demonstrate higher recognition rates, underscoring their more pronounced facial expressions. This stability indicates that the facial expressions associated with these emotions are more discernible, even when features are partially occluded. The emergence of surprise as a leading emotion under mask occlusion highlights the adaptability of emotion recognition models when faced with occluded features. This suggests

that the models may begin to rely more on specific facial characteristics or contexts that remain visible during occlusion, such as raised eyebrows and wide-open eyes.Neutral and happiness maintain relatively high recognition rates across scenarios, emphasizing the clear and distinct facial expressions that facilitate recognition under adverse conditions.

In terms of overall accuracy using the FERV39K dataset, ResNet18 achieved the highest accuracy in the no occlusion scenario with 94.0%, followed by VGG19 at 93.3%, and EfficientNetB1 at 88.3% (Table 4.2a).

ResNet18 performed better than the others when simulating occlusion of VR goggles with an accuracy of 80.9%, followed by VGG19 with 80.8% and EfficientNetB1 with 77.3%, as shown in the Table 4.2b. Under mask occlusion conditions, VGG19 led with 80.8%, followed by ResNet18 at 80.6% and EfficientNetB1 at 77.8% (Table 4.2c).

It is evident that, in the initial results without occlusion, the performance of the model in the FERV39K dataset follows the same trend observed in the FERPlus dataset, where ResNet18 achieves the highest accuracy, followed by VGG19 and then EfficientNet. In occlusion scenarios, the distinctions in dataset structure and model robustness become more evident, which is why this ranking does not hold in comparison with the results obtained by the models in FERPlus dataset. Although EfficientNetB1 reveals effective, it did not handle sequential data of FER39K as effectively as models like ResNet18 or VGG19. In contrast, FERPlus, composed of static images, shows to align better with the capabilities of EfficientNet, resulting in improved performance.

In a more in-depth analysis, the no-occlusion scenario revealed that ResNet18 had the lowest recognition rates for surprise and sadness, both at 93.7%, and for happiness at 90.6%. Conversely, it achieved the highest recognition rates for disgust at 98.1% and fear at 97.6%. For VGG19, the top recognized emotions were disgusted (98.2%) and fear (97.6%), while the poorest performances occurred in sadness (90.3%) and happiness (92.1%). EfficientNetB1 also identified disgust (98.5%) and fear (95.3%) as leading emotions, mirroring ResNet18 results. However, it had the lowest recognition rates for happiness (78.3%) and sadness (85.6%), similar to VGG19 performance.

When upper occlusion was introduced, ResNet18 exhibited the lowest recognition rates

Table 4.1: Accuracy per class in FERPlus dataset

(a) No occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Neutral	0,861	0,861	0,865
Happiness	0,923	0,923	0,929
Surprise	0,856	0,865	0,865
Sadness	0,616	0,662	0,636
Anger	0,705	0,768	0,685
Disgust	0,476	0,428	0,428
Fear	0,366	0,400	0,383
Contempt	0,280	0,240	0,240
Accuracy	**0,861**	0,830	0,823

(b) Goggles occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Neutral	0,823	0,836	0,847
Happiness	0,899	0,911	0,923
Surprise	0,818	0,821	0,825
Sadness	0,498	0,586	0,509
Anger	0,632	0,666	0,700
Disgust	0,476	0,523	0,428
Fear	0,300	0,383	0,350
Contempt	0,280	0,200	0,240
Accuracy	0,773	0,795	**0,797**

(c) Mask occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Neutral	0,725	0,713	0,760
Happiness	0,752	0,801	0,740
Surprise	0,787	0,812	0,815
Sadness	0,536	0,521	0,555
Anger	0,632	0,570	0,652
Disgust	0,380	0,380	0,476
Fear	0,300	0,400	0,333
Contempt	0,200	0,200	0,200
Accuracy	0,695	0,704	**0,712**

for neutral at 28.8% and for anger at 95.2%. The highest accuracies were observed in happiness at 97.5% and fear at 97.2%. For VGG19, neutral remained the least recognized emotion at 31.7%, alongside anger at 93.8%. The highest accuracies were achieved in fear at 97.2%, consistent with the previous scenario, and happiness at 96.9%. In the case of EfficientNetB1, neutral also had the lowest accuracy at 27.4%, followed by anger at 88.3%. The highest accuracies were noted in sadness at 95.4% and happiness at 94.5% (the opposite of the no-occlusion scenario).

In the scenario involving mask occlusion, ResNet18 continued to show the lowest recognition rates for neutral at 28.9% and for surprise at 94.3%. The highest accuracies were achieved in happiness at 96.8% and fear at 96.1%, mirroring the results from the VR goggles occlusion scenario. For VGG19, neutral remained the least recognized emotion at 30.4%, followed by surprise at 94.1%. The highest accuracies were noted in happiness at 96.5%, consistent with the VR occlusion scenario, and anger at 95.8%. In the case of EfficientNetB1, the trend persisted with neutral as the worst emotion at 32.9% accuracy, followed by surprise at 88.5%. The highest accuracies were recorded in fear at 94.5%, consistent with the first scenario, and happiness at 92.3%.

Significant inconsistencies in recognition rates across various occlusion scenarios are revealed by the FERV39K dataset, which features sequential images that capture dynamic emotions. Neutral emotions always had the lowest recognition rates in bot occlusion scenarios. The introduction of occlusions, whether through upper occlusion or mask occlusion, likely contributed to the disruption of facial features that convey subtle cues. The lowest recognition rates for emotions such as surprise and sadness across different scenarios indicate that they are particularly prone to occlusions. When facial features are obscured, it can be harder to detect sadness when it involves less pronounced facial movement. The recognition rates for emotions like happiness and fear fluctuated depending on the specific occlusion scenarios. The recognition rate for happiness in the no-occlusion scenario was lower for EfficientNetB1, possibly showing that the model relies on more visible facial cues for accurate identification. Introducing occlusions led to an increase in both sadness and happiness recognition rates for EfficientNetB1. The model may have

identified features that are less affected by occlusions, which is why the improvement in sadness recognition rates from the no-occlusion scenario to the occlusion scenario suggests better performance.

These inconsistencies could be exacerbated by the representation of emotions in the FERV39K dataset. Emotions like happiness and fear tend to involve more distinctive facial expressions, making them more recognizable under certain conditions. In contrast, neutral expressions and emotions such as sadness may lack distinctiveness, leading to variability in recognition accuracy across scenarios.

Table 4.2: Accuracy per class in FERV39K dataset

(a) No occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Neutral	0,948	0,944	0,905
Happiness	0,906	0,921	0,783
Surprise	0,937	0,934	0,917
Sadness	0,937	0,903	0,856
Anger	0,947	0,935	0,914
Disgust	0,981	0,982	0,985
Fear	0,976	0,976	0,953
Accuracy	**0,940**	0,933	0,883

(b) Goggles occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Neutral	0,288	0,317	0,274
Happiness	0,975	0,969	0,945
Surprise	0,963	0,959	0,902
Sadness	0,971	0,951	0,954
Anger	0,952	0,938	0,883
Disgust	0,959	0,958	0,900
Fear	0,972	0,972	0,935
Accuracy	**0,809**	0,808	0,773

(c) Mask occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Neutral	0,289	0,304	0,329
Happiness	0,968	0,965	0,923
Surprise	0,943	0,941	0,885
Sadness	0,959	0,953	0,917
Anger	0,952	0,958	0,893
Disgust	0,954	0,956	0,902
Fear	0,961	0,957	0,945
Accuracy	0,806	**0,808**	0,778

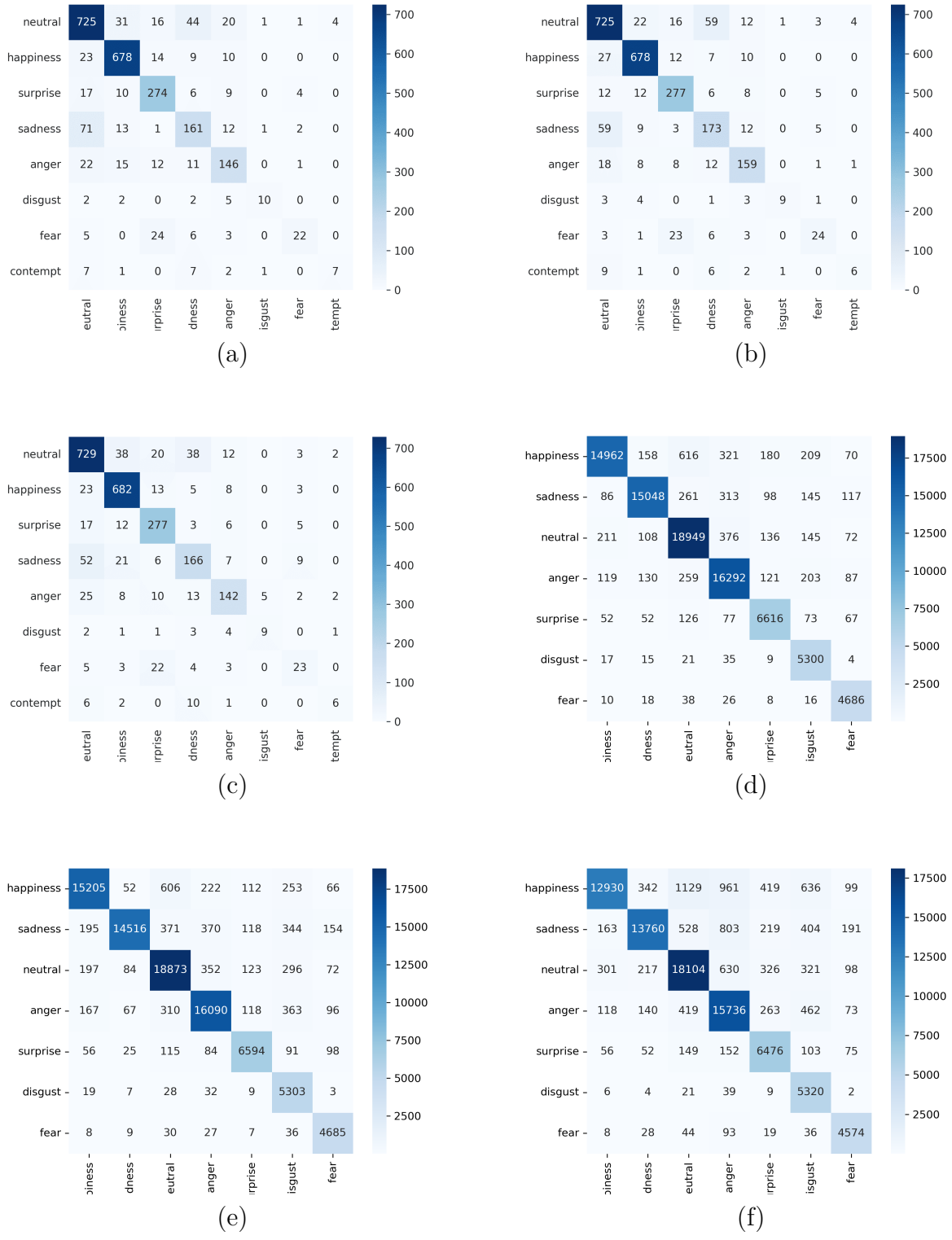


Figure 4.1: Confusion Matrices obtained using FERPlus and FERV39K datasets in no occlusion scenario. **(a)** ResNet18; **(b)** VGG19; **(c)** EfficientNetB1; **(d)** ResNet18 for FERV39K; **(e)** VGG19 for FERV39K; **(f)** EfficientNetB1 for FERV39K.

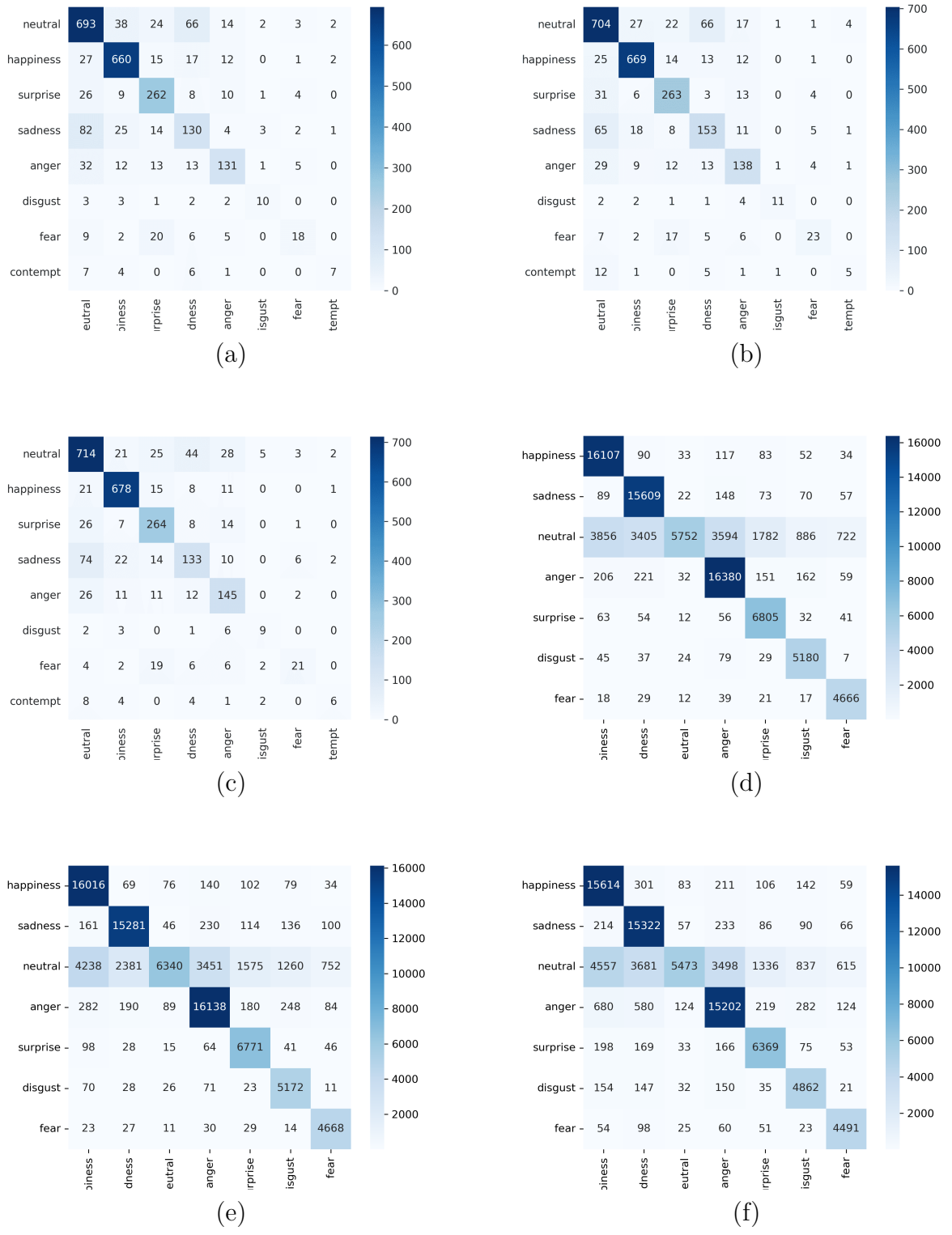


Figure 4.2: Confusion Matrices obtained using FERPlus and FERV39K datasets in upper occlusion scenario. **(a)** ResNet18; **(b)** VGG19; **(c)** EfficientNetB1; **(d)** ResNet18 for FERV39K; **(e)** VGG19 for FERV39K; **(f)** EfficientNetB1 for FERV39K.

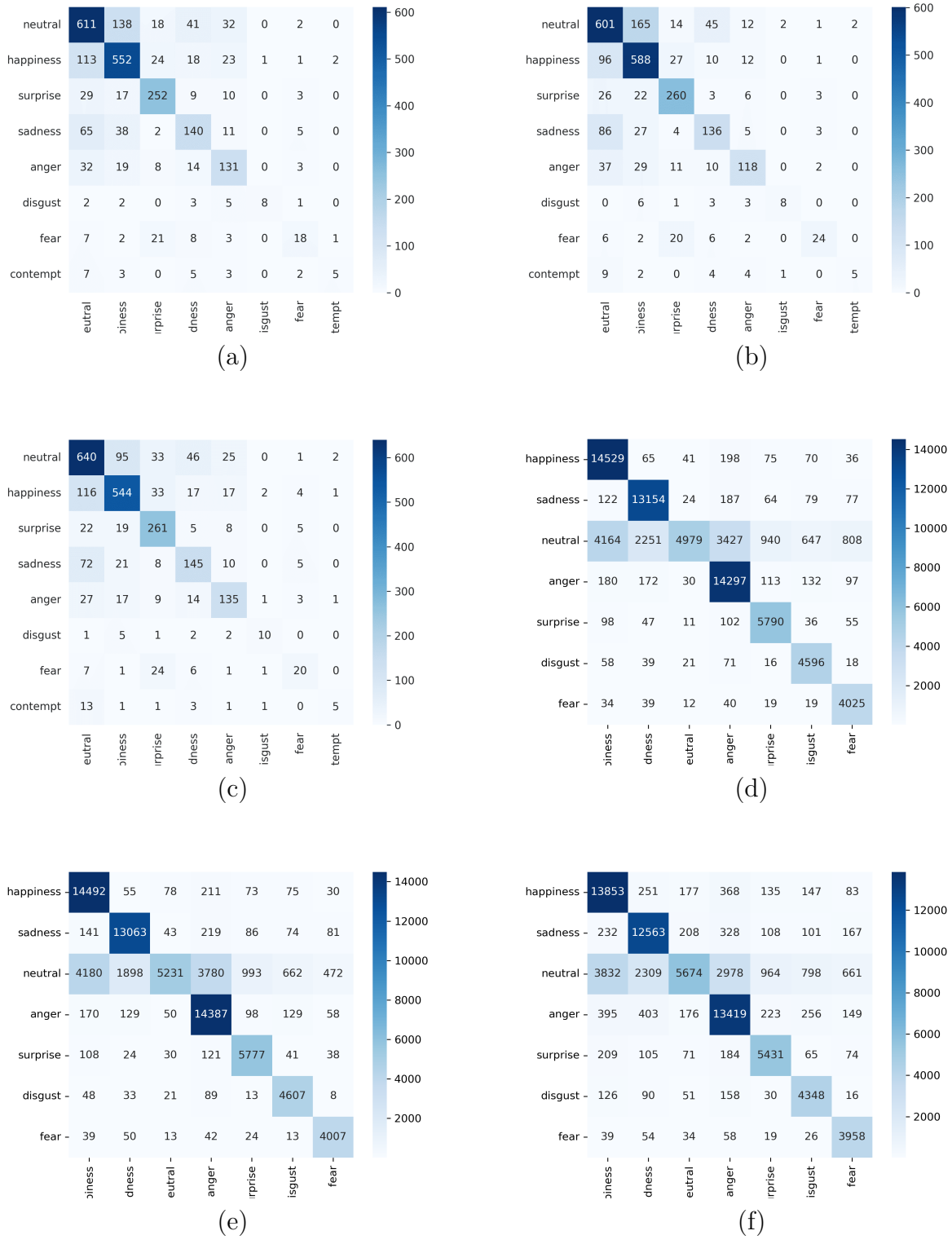


Figure 4.3: Confusion Matrices obtained using FERPlus and FERV39K datasets in mask occlusion scenario. **(a)** ResNet18; **(b)** VGG19; **(c)** EfficientNetB1; **(d)** ResNet18 for FERV39K; **(e)** VGG19 for FERV39K; **(f)** EfficientNetB1 for FERV39K.

The subsequent overview details the performance results for each class grouping across the multiple models used in this research.

Regarding overall accuracy on the FERPlus dataset without any occlusion, VGG19 attained the top accuracy of 84.5%, with EfficientNetB1 achieving 83.8%, and ResNet18 scoring 82.8% (see Table 4.3). In every model, the highest accuracy was achieved by the positive class, scoring 92.7% in VGG19, 89.9% in ResNet18, and 88.7% in EfficientNetB1. In contrast, the negative expressions showed lower accuracy, at 76.1% with EfficientNetB1, 72.5% with VGG19, and 71.6% with ResNet18.

In goggles occlusion scenario, the top model maintains the same, VGG19, with 81.0% of accuracy, followed by EfficientNetB1 with 80.6% and ResNet18 with 79.9%. Positive class remains the class with the highest accuracy, 88.9% in VGG19, 88.0% in EfficientNetB1 and 87.0% in ResNet18. Negative class maintains the lower accuracy, with 75.3% in VGG19, 71.3% in ResNet18 and 68.0% in EfficientNetB1.

Table 4.3: Accuracy per class using class grouping in FERPlus dataset

(a) No occlusion

Class	**VGG19**	**ResNet18**	**EfficientNetB1**
Negative	0,725	0,716	0,761
Positive	0,927	0,899	0,887
Neutral	0,847	0,835	0,842
Accuracy	**0,845**	0,828	0,838

(b) Goggles occlusion

Class	**VGG19**	**ResNet18**	**EfficientNetB1**
Negative	0,753	0,713	0,680
Positive	0,889	0,870	0,880
Neutral	0,765	0,788	0,820
Accuracy	**0,810**	0,799	0,806

Regarding the class grouping, using FERV39K, in the no occlusion scenario, VGG19 achieved the highest accuracy with 82.5%, followed by ResNet18 with 82.4% and EfficientNetB11 with 80.7% (Table 4.4a).

Upon conducting a more comprehensive analysis, it was found that negative expression exhibited the highest performance levels in both ResNet18 (98.3%) and VGG19 (97.3%). Conversely, within EfficientNetB1, the positive expression showed the best results, achieving 96.1%. Clearly, the neutral expression consistently delivered the lowest performance across all models, with outcomes of 29.3% in VGG19, 31.4% in ResNet18, and 43.7% in EfficientNetB1.

Upon introducing upper occlusion (Table 4.4b), ResNet18 attained the highest accuracy at 81.4%, followed closely by VGG19 at 81.2%, and EfficientNetB1 at 79.5%. In a deeper examination, positive expression exhibited peak performance with ResNet18 reaching 97.4%, VGG19 achieving 97.5%, and EfficientNetB1 at 96.6%. Conversely, neutral expression showed the least performance across the models, with ResNet18 at 28.9%, VGG19 at 28.6%, and EfficientNetB1 at 24.9%.

Table 4.4: Accuracy per class using class grouping in FERV39K dataset

(a) No occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Negative	0,983	0,973	0,837
Positive	0,981	0,980	0,961
Neutral	0,293	0,314	0,437
Accuracy	0,824	**0,825**	0,807

(b) Goggles occlusion

Class	**ResNet18**	**VGG19**	**EfficientNetB1**
Negative	0,964	0,959	0,944
Positive	0,974	0,975	0,966
Neutral	0,289	0,286	0,249
Accuracy	**0,814**	0,812	0,795

In the analysis of confusion matrices for the FERPlus and FERV39K datasets, model performance varies under normal, mask occlusion, and goggles occlusion conditions, with class grouping revealing distinct strengths and weaknesses in emotion recognition. Under normal conditions (Figure 4.1), all models demonstrate reliable accuracy in distinguishing

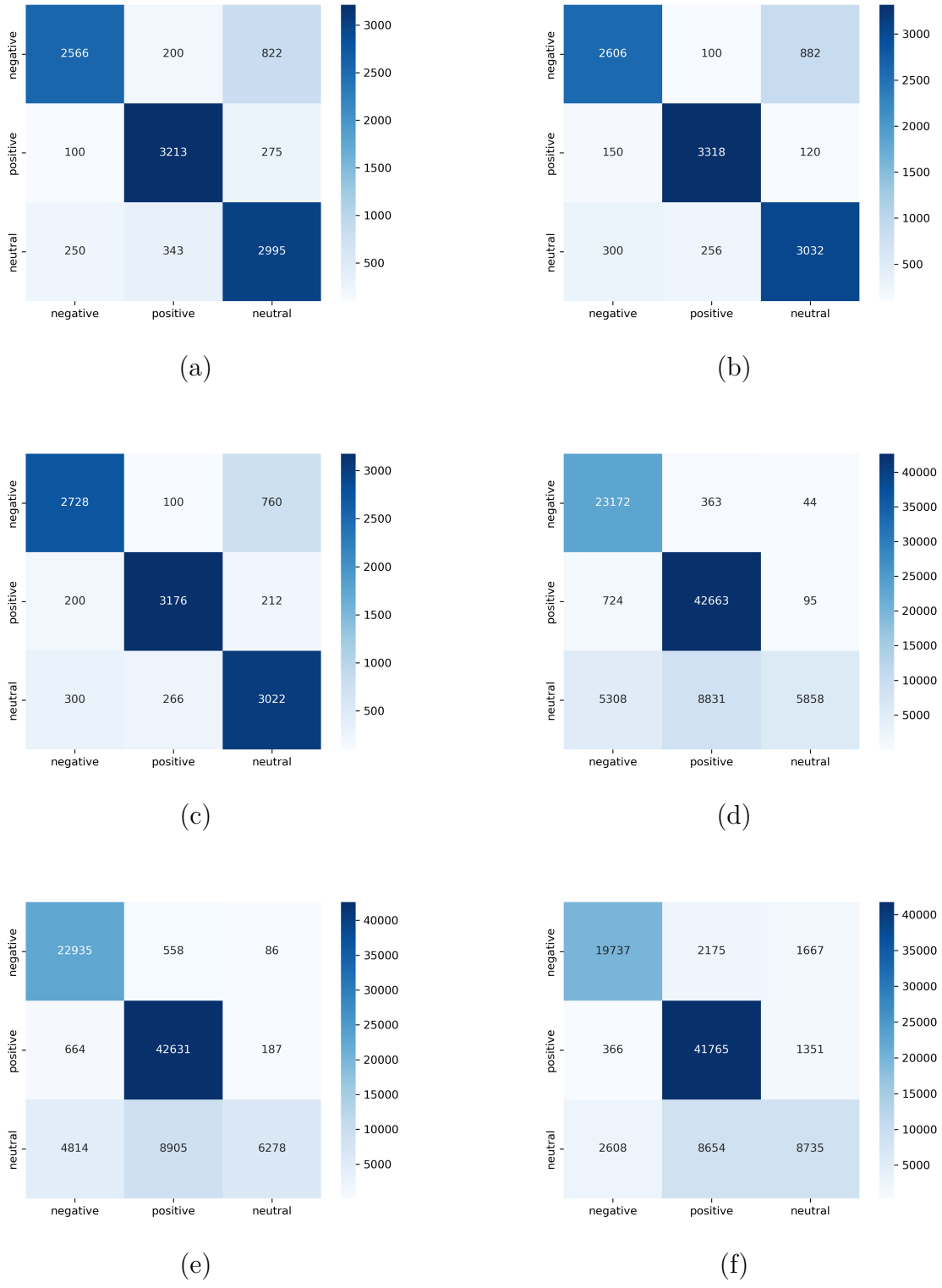


Figure 4.4: Confusion Matrices obtained using FERPlus and FERV39K datasets in no occlusion scenario using class grouping. **(a)** ResNet18; **(b)** VGG19; **(c)** EfficientNetB1; **(d)** ResNet18 for FERV39K; **(e)** VGG19 for FERV39K; **(f)** EfficientNetB1 for FERV39K.

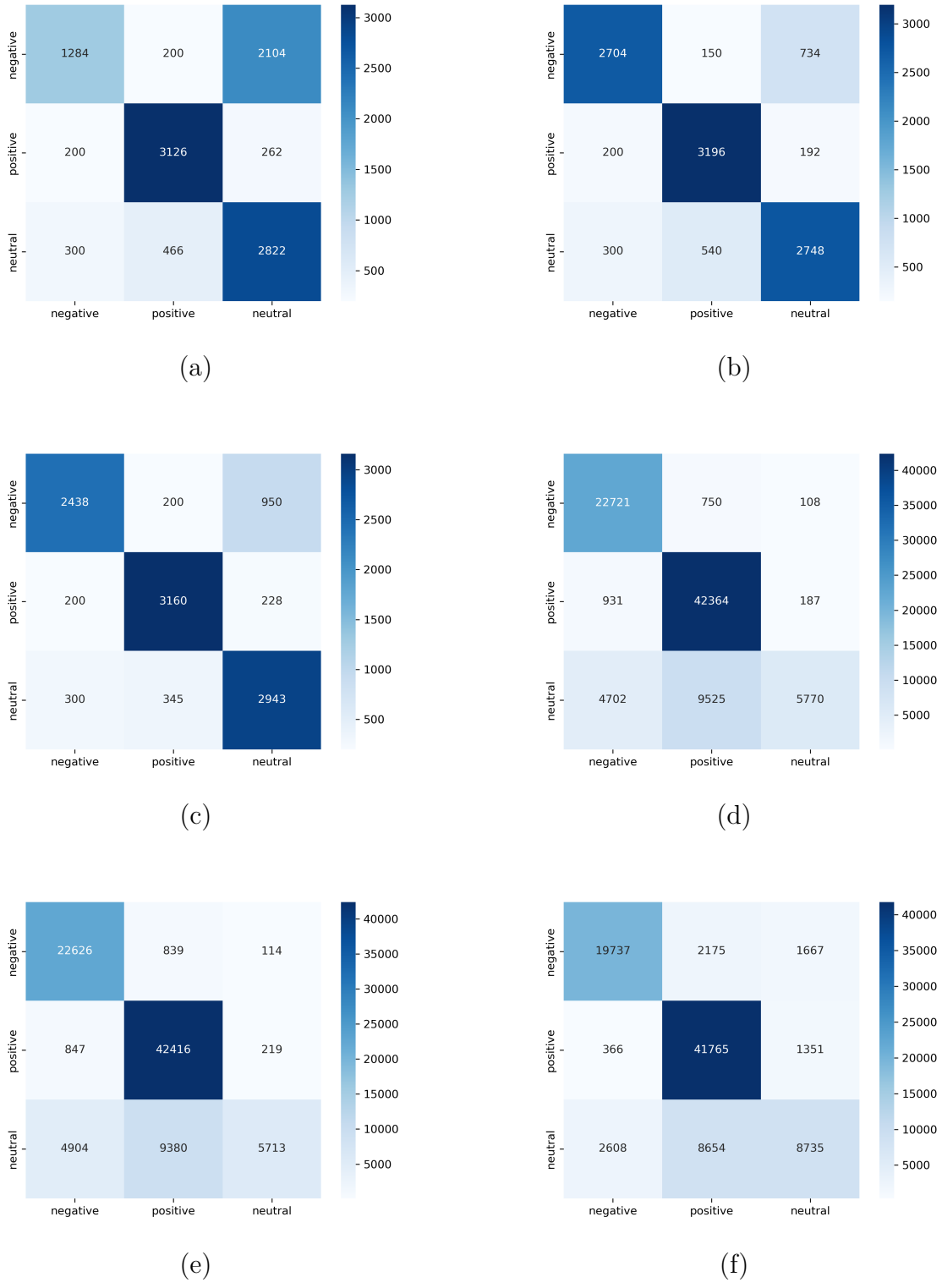


Figure 4.5: Confusion Matrices obtained using FERPlus and FERV39K datasets in upper occlusion scenario using class grouping. **(a)** ResNet18; **(b)** VGG19; **(c)** EfficientNetB1; **(d)** ResNet18 for FERV39K; **(e)** VGG19 for FERV39K; **(f)** EfficientNetB1 for FERV39K.

emotions such as happiness and sadness, while showing some misclassifications in neutral expressions, particularly among emotions like fear and contempt, which often share similar visual cues. Mask occlusion (Figure 4.3) significantly reduces accuracy, especially for emotions that rely on lower-face visibility, such as anger and disgust, leading to misclassifications where these emotions are often confused with neutral or positive expressions. Goggles occlusion (Figure 4.2) primarily impacts the detection of expressions like sadness and surprise, which depend heavily on eye visibility, resulting in a higher rate of confusion within the neutral, happiness and surprise. Class grouping (Figure 4.4 and Figure 4.5) reveals that positive emotions, like happiness, are generally the easiest to classify accurately across all conditions, while neutral expressions and negative expressions show increased misclassification rates, indicating the challenges posed by occlusion in facial expression recognition.

On the FERV39K dataset, ResNet18 did not perform as strongly as it did on FERPlus. The results revealed that the model faced difficulties in correctly classifying certain emotions, especially within the negative and neutral categories. Although ResNet18 surpassed VGG19 on the FERV39K dataset, it lagged behind EfficientNetB1. The performance of VGG19 on FERV39K was noticeably inferior compared to its results on FERPlus, as it struggled with distinguishing between various emotion classes, which pointed to challenges in identifying subtle emotional expressions. Both ResNet18 and EfficientNetB1 outperformed VGG19 on FERV39K. EfficientNetB1 excelled on this dataset, achieving the highest accuracy by effectively reducing confusion among different emotion classes. Its proficiency at extracting distinctive features from the FERV39K dataset was reflected in its superior results. Overall, all three models showed diminished performance when moving from the FERPlus to the FERV39K dataset, indicating that FERV39K was more challenging for these models. EfficientNetB1 consistently outshone ResNet18 and VGG19 across both datasets, highlighting the advantage of its architecture in capturing relevant features for emotion classification. An examination revealed that all models had difficulty with certain emotions, especially the negative and neutral classes, on both datasets. The FERV39K dataset appeared to be more challenging due to its larger number of images

and the presence of sequential emotion-depicting images. This sequence made recognition more difficult as emotions could appear more alike or be more subtle compared to those in FERPlus. The significant drop in the performance of ResNet18 on FERV39K implied that it might be less capable of managing the complexities of the dataset. VGG19, in particular, faced a stark decline when transitioning to FERV39K, suggesting its architecture was not as well-equipped for the challenges of the dataset. Conversely, EfficientNetB1 retained its outstanding performance on both datasets, demonstrating its robust ability to handle diverse datasets and the challenges of FERV39K effectively.

4.2 Discussion

Occlusion clearly impacts the capacity of the models to discern emotions, with the effects varying by which facial region is obscured. Both FERPlus and FERV39K exhibit marked declines in accuracy due to occlusion, though the specific area covered critically influences the extent of this drop. In FERPlus, the absence of temporal data exacerbates the impact of missing facial features. For instance, eye occlusion reduces the accuracy of VGG19 to 79.5% and EfficientNetB1 to 79.7%. In contrast, when a mask covers the mouth and nose, the accuracy of EfficientNetB1 sharply fall to 71.2%, highlighting the importance of the lower face, especially for emotions like happiness, often conveyed through smiles. Meanwhile, in FERV39K, temporal sequences slightly alleviate occlusion effects, but significant performance losses still occur. Under goggles occlusion, the EfficientNetB1 score decreased from 88.3% to 77.3%, indicating that even temporal data can not fully compensate for the obstruction of key features like the eyes. Emotions reliant on eye expressions, such as fear and sadness, are notably difficult to detect with goggles; for example, fear recognition falls to 30% for ResNet18 in FERPlus compared to 36.6% without occlusion. Likewise, expressions of happiness, dependent on mouth visibility, are heavily impacted by mask occlusion, with the accuracy of EfficientNetB1 dropping from 92.9 % to 74%. While dynamic datasets like FERV39K benefit somewhat from temporal cues, occluding essential facial areas like the eyes and mouth still leads to pronounced performance drops.

Organizing specific emotions into broader categories like negative, positive, and neutral simplified the process, thereby boosting model performance by reducing complexity. This technique proved highly effective in the FERPlus dataset, where VGG19 attained an accuracy of 84.5%, ResNet18 achieved 82.8%, and EfficientNetB1 reached 84.5% without occlusion when employed with class grouping. These outcomes outperform detailed per-class identification, where distinguishing between particular emotions such as fear and disgust is more difficult. In FERV39K, class grouping continued to be beneficial, even with the inclusion of temporal data. For instance, ResNet18 achieved 82.4% accuracy in FERV39K with grouped classes and no occlusion, demonstrating that merging temporal data with class grouping improves model stability compared to static data alone. The use of three-class grouping provides benefits in both datasets, but it is especially impactful in occlusion scenarios. Grouping emotions simplified the classification task and improved robustness, even when facial features are occluded. For example, in the FERPlus dataset under goggles occlusion, VGG19 achieved an accuracy of 79.5% with grouped classes, which is higher than its per-class performance under the same condition. Similarly, ResNet18 in FERV39K maintained 82.4% accuracy with class grouping and no occlusion, demonstrating that this strategy is effective even when dealing with dynamic sequences.

A significant observation is the notable disparity in model performance regarding neutral emotion detection across the two datasets. In the case of FERV39K, the evolving characteristics of sequences introduced variability in the expression of neutrality. Over time, subtle facial movements could be perceived by the model as emotional changes, resulting in increased misclassifications. This challenge caused reduced accuracy for neutral emotions in FERV39K, as the model found it difficult to consistently detect subtle, less expressive emotions. For instance, minimal movements like a slight eyebrow lift or a minor shift in mouth position might be seen as transitions between emotional states, potentially confusing the model and leading to lower accuracy for neutral emotions in dynamic sequences. On the other hand, FERPlus features static images that capture a single moment of facial expression, simplifying the task for the model to recognize and

categorize neutral expressions. Under these static conditions, neutral emotions were more clearly delineated, leading to improved accuracy.

In summary, class grouping significantly improves model performance by simplifying the classification task, especially in FERPlus with its static images. However, the use of dynamic sequences in FERV39K adds robustness, particularly in occlusion scenarios, as temporal cues provide additional context. Despite this, occluding key facial regions like the eyes or mouth still causes significant performance drops, especially for emotions that rely on these features. Finally, the Neutral emotion presents a unique challenge in FERV39K, where small facial movements in dynamic sequences can confuse the model, leading to lower accuracy compared to the more static, well-defined Neutral expressions in FERPlus.

4.3 Summary

Three CNNs: VGG19, ResNet18, and EfficientNetB1 were used in the study to investigate how facial emotion recognition systems respond to occlusions, such as masks, VR goggles, or other facial coverings. Testing of these models was conducted on two different datasets: FERPlus, which contains static images, and FERV39K, which incorporates sequential images that add valuable temporal information for dynamic settings. The main conclusion of the research is that occlusions consistently decrease recognition accuracy in all models and datasets. Regardless of the type or size of the occlusion, this effect was seen, which indicates a significant weakness in CNN-based systems when they come across obstructed facial features. The amount of performance degradation can be affected by the specific location of the occlusion on the face. For instance, occluding the eye region hinders the recognition of emotions like fear and sadness, while covering the mouth and nose primarily affects the recognition of happiness and surprise, which rely heavily on visible mouth movements.Differences in model performance were also evident. In the FERPlus dataset, ResNet18 was superior to VGG19 and EfficientNetB1 in scenarios without occlusion, indicating its superiority in static image analysis. EfficientNetB1 had an advantage

over the other models in identifying subtle emotional shifts, while VGG19 had trouble distinguishing emotions with low-intensity expressions. These variations in performance highlight how the sequential and larger FERV39K dataset posed a greater challenge to the models, especially when compared to the more straightforward static classification tasks in the FERPlus dataset. In addition, it was found that grouping emotions into broader categories, such as positive, negative, and neutral, improved classification accuracy across all models. This approach reduced the cognitive load on the models and improved accuracy by simplifying the classification task, especially when partial occlusions were present. This suggests a potential path to enhance model performance when fine-grained emotion classification is not required. Further investigation revealed challenges and patterns of robustness that were specific to emotions. Certain emotions, such as contempt and fear, were notably difficult to recognize when occlusion was present. The emotional states are heavily influenced by subtle facial cues around the eyes and mouth, regions that are often obscured by occlusions. This finding emphasizes the difficulty that current CNN models experience in generalizing across missing facial information, especially for emotions that do not have distinct facial markers. In contrast, emotions such as happiness and neutrality were relatively more resilient to occlusion. Happiness, for example, is often associated with a larger, more distinctive smile, while neutral expressions are characterized by the absence of particular emotional markers, making them less vulnerable to performance degradation when portions of the face are obscured.

Chapter 5

Conclusion

Although there have been notable improvements in FER technologies, considerable difficulties remain, especially regarding partial occlusion, which can obstruct the precise interpretation of emotions. These obstructions block essential facial features, making it challenging to detect important emotional cues, such as eye dynamics and mouth movements.

When evaluating performance, the findings revealed substantial differences under diverse conditions. Under optimal scenarios without any obstruction, ResNet18 achieved the highest level of accuracy on both the FERPlus and FERV39K datasets, outperforming VGG19 and EfficientNetB1. This demonstrates its strong capability to accurately identify and categorize facial expressions when all features are visible. Occlusion drastically decreased models performance; for example, when a mask covered the mouth, the accuracy of EfficientNetB1 dropped significantly, in happiness, from 92.9% to 74%, in FERPlus dataset. Moreover, in FERV39K, occlusion scenarios significantly impact the accuracy of the neutral class. For instance, in VGG19, the accuracy drops drastically from 94.4% to 31.7% in the goggles occlusion scenario and to 30.4% in the mask occlusion scenario. Emotions like contempt and fear were challenging to recognize under occlusion, as they rely on subtle facial cues around the eyes and mouth, which are often obscured. This highlighted the difficulty CNN models face in generalizing with missing facial information, especially for emotions lacking distinct markers. In contrast, emotions like happiness

and neutrality proved more resilient, as happiness is marked by a distinctive smile, and neutrality lacks strong emotional features, making them less affected by occlusion.

Implementing a strategy that groups classes into three categories improved the effectiveness of the models, especially in scenarios with occlusion. For instance, when employing grouped classes, VGG19 achieved an accuracy of 81.0% with goggles occlusion, in FERPlus, showing a notable enhancement compared to its performance using separate classes. This approach streamlined the classification process, allowing the models to achieve superior performance even in the presence of obstructed features.

The complexity of extracting distinctive features from occluded facial areas can result in inaccurate feature localization, imprecise facial alignment, and errors in facial registration. Consequently, emotions that rely heavily on the visibility of certain facial features become more difficult to identify accurately. Therefore, addressing the parameters of occlusion is essential for enhancing the effectiveness of FER systems. To enhance the effectiveness of facial expression recognition systems, several avenues for future work are proposed. First, developing specialized datasets that feature scenarios with masks or VR goggles is crucial. These datasets should reflect real-world circumstances where individuals might wear masks or goggles, providing a valuable resource for training models to manage occlusions efficiently.To enhance the models ability to recognize emotions under difficult circumstances, researchers must include a diverse range of facial expressions and demographic variations in their datasets. Besides building these datasets, the application of data augmentation techniques can significantly strengthen model robustness. For example, synthetic occlusion, which intentionally covers portions of the face during training, helps models better generalize to unforeseen occlusions encountered in real-world applications. Moreover, investigating different model architectures focused on temporal features could benefit greatly from these tailored datasets. Training models like 3D CNNs or RNNs on sequences that include occlusions can foster strong representations for emotion recognition even with partial face visibility. Addressing these aspects could considerably progress the field of facial expression recognition, leading to more precise and practical models for everyday use.

Bibliography

[1] D. Matsumoto, M. Frank, and H. Hwang, *Nonverbal Communication: Science and Applications: Science and Applications* (EBSCO ebook academic collection). SAGE Publications, 2013, ISBN: 978-1-4129-9930-4. [Online]. Available: https://books.google.pt/books?id=PeOeu3qFFTIC.

[2] A. A. Pise, M. A. Alqahtani, P. Verma, *et al.*, "Methods for Facial Expression Recognition with Applications in Challenging Situations," *Computational Intelligence and Neuroscience*, vol. 2022, p. 9 261 438, May 2022. DOI: 10.1155/2022/9261438.

[3] C. E. Izard, Ed., *Human emotions* (Emotions, personality, and psychotherapy). New York: Plenum Press, 1977, ISBN: 978-0-306-30986-1.

[4] S. Saurav, A. Saini, R. Saini, and S. Singh, "Deep learning inspired intelligent embedded system for haptic rendering of facial emotions to the blind," English, *Neural Computing and Applications*, vol. 34, no. 6, pp. 4595–4623, 2022, Publisher: Springer Science and Business Media Deutschland GmbH, ISSN: 09410643. DOI: 10.1007/s00521-021-06613-3. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85117830803&doi=10.1007%2fs00521-021-06613-3&partnerID=40&md5=ff31d4bd7bc4190b4483b813b2837a34.

[5] S. Singh, A. Gupta, and R. Pavithr, "Automatic classroom monitoring system using facial expression recognition," English, in *Lecture Notes in Electrical Engineering*, G. Sanyal, C. Travieso-Gonzalez, S. Awasthi, C. Pinto, and B. Purushothama, Eds., vol. 836, Scopus ID: 2-s2.0-85130262593, Germany: Springer Science and Business Media Deutschland GmbH, 2022, pp. 151–165, ISBN: 9789811685415. DOI: 10.1007/

978-981-16-8542-2_12. [Online]. Available: https://doi.org/10.1007/978-981-16-8542-2_12.

[6] X. Zhao and S. Zhang, "A Review on Facial Expression Recognition: Feature Extraction and Classification," en, *IETE Technical Review*, vol. 33, no. 5, pp. 505–517, Sep. 2016, ISSN: 0256-4602, 0974-5971. DOI: 10.1080/02564602.2015.1117403. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/02564602.2015.1117403 (visited on 06/27/2022).

[7] P. Naveen, "Occlusion-aware facial expression recognition: A deep learning approach," *Multimedia Tools and Applications*, Sep. 2023, ISSN: 1573-7721. DOI: 10.1007/s11042-023-17013-1.

[8] B. Houshmand and N. Mefraz Khan, "Facial Expression Recognition Under Partial Occlusion from Virtual Reality Headsets based on Transfer Learning," in *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*, 2020, pp. 70–75. DOI: 10.1109/BigMM50055.2020.00020.

[9] J. Brownlee, *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery, 2019. [Online]. Available: https://books.google.pt/books?id=DOamDwAAQBAJ.

[10] H. Yu, "Facial expression recognition with computer vision," *Applied and Computational Engineering*, vol. 37, pp. 74–80, Feb. 2024. DOI: 10.54254/2755-2721/37/20230473.

[11] K. Vinutha, M. Kumar Niranjan, J. Makhijani, B. Natarajan, V. Nirmala, and T. R. Vijaya Lakshmi, "A Machine Learning based Facial Expression and Emotion Recognition for Human Computer Interaction through Fuzzy Logic System," in *2023 International Conference on Inventive Computation Technologies (ICICT)*, ISSN: 2767-7788, Apr. 2023, pp. 166–173. DOI: 10.1109/ICICT57646.2023.10134493.

[12] R. P. Lopes, B. Barroso, L. Deusdado, *et al.*, "Digital Technologies for Innovative Mental Health Rehabilitation," en, *Electronics*, vol. 10, no. 18, p. 2260, Jan. 2021,

Number: 18 Publisher: Multidisciplinary Digital Publishing Institute, ISSN: 2079-9292. DOI: 10.3390/electronics10182260.

[13] L. Liu, "Application of facial expression recognition based on domain-adapted convolutional neural network in English smart teaching system," en, *Soft Computing*, vol. 27, no. 12, pp. 8437–8448, Jun. 2023, ISSN: 1433-7479. DOI: 10.1007/s00500-023-08143-7.

[14] P. C. Sánchez and C. C. Bennett, "Facial expression recognition via transfer learning in cooperative game paradigms for enhanced social AI," en, *Journal on Multimodal User Interfaces*, vol. 17, no. 3, pp. 187–201, Sep. 2023, ISSN: 1783-8738. DOI: 10.1007/s12193-023-00410-z.

[15] Z. Shi, *Advanced Artificial Intelligence* (Series on intelligence science). World Scientific, 2011, ISBN: 9789814291347. [Online]. Available: https://books.google.pt/books?id=wNbMOoTuGU0C.

[16] Z. Cai, L. Liu, B. Chen, and Y. Wang, *Artificial Intelligence: From Beginning To Date*. World Scientific Publishing Company, 2021, ISBN: 9789811223730. [Online]. Available: https://books.google.pt/books?id=x30xEAAAQBAJ.

[17] Y. Jiang, X. Li, H. Luo, S. Yin, and O. Kaynak, "Quo vadis artificial intelligence?" *Discover Artificial Intelligence*, vol. 2, no. 1, p. 4, 2022.

[18] E. Strickland, "Ibm watson, heal thyself: How ibm overpromised and underdelivered on ai health care," *IEEE Spectrum*, vol. 56, no. 4, pp. 24–31, 2019. DOI: 10.1109/MSPEC.2019.8678513.

[19] Z. Khan, "Ai in surgery: Robotics and beyond,"

[20] M. C.-T. Tai, "The impact of artificial intelligence on human society and bioethics," *Tzu chi medical journal*, vol. 32, no. 4, pp. 339–343, 2020.

[21] A. Senthilselvi, B. Chelliah, and S. Pandi, *Machine Learning*. Shanlax Publications, 2021, ISBN: 9789391373856. [Online]. Available: https://books.google.pt/books?id=vUpgEAAAQBAJ.

[22] J. Mueller and L. Massaron, *Machine Learning For Dummies* (For dummies). Wiley, 2016, ISBN: 9781119245513. [Online]. Available: https://books.google.pt/books?id=JLEyDAAAQBAJ.

[23] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1–6. DOI: 10.1109/ICCUBEA.2018.8697857.

[24] K. Gurney, *An Introduction to Neural Networks*. CRC Press, 2018, ISBN: 9781482286991. [Online]. Available: https://books.google.pt/books?id=e0pZDwAAQBAJ.

[25] A. Krenker, J. Bešter, and A. Kos, "Introduction to the artificial neural networks," *Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech*, pp. 1–18, 2011.

[26] M. Ekman, *Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow*. Pearson Education, 2021, ISBN: 9780137470297. [Online]. Available: https://books.google.pt/books?id=wNnPEAAAQBAJ.

[27] J. D. Kelleher, *Deep learning*. MIT press, 2019.

[28] K. Hong, S. K. Chalup, and R. A. King, "A component based approach for classifying the seven universal facial expressions of emotion," in *2013 IEEE Symposium on Computational Intelligence for Creativity and Affective Computing (CICAC)*, 2013, pp. 1–8. DOI: 10.1109/CICAC.2013.6595214.

[29] L. Mozaffari, M. M. Brekke, B. Gajaruban, D. Purba, and J. Zhang, "Facial Expression Recognition Using Deep Neural Network," in *2023 3rd International Conference on Applied Artificial Intelligence (ICAPAI)*, IEEE, 2023, pp. 1–9.

[30] D. Canedo and A. Neves, "Mood estimation based on facial expressions and postures," in *Proceedings of the RECPAD*, 2020, pp. 49–50.

[31] N. Petrou, G. Christodoulou, K. Avgerinakis, and P. Kosmides, "Lightweight Mood Estimation Algorithm For Faces Under Partial Occlusion," in *Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments*, 2023, pp. 402–407.

[32] C. P. Loizou, "An automated integrated speech and face imageanalysis system for the identification of human emotions," en, *Speech Communication*, vol. 130, pp. 15–26, Jun. 2021, ISSN: 01676393. DOI: 10.1016/j.specom.2021.04.001. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S016763932100039X (visited on 10/13/2021).

[33] Z. Liu, M. J. Zuo, and H. Xu, "Parameter selection for Gaussian radial basis function in support vector machine classification," en, in *2012 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, Chengdu, China: IEEE, Jun. 2012, pp. 576–581, ISBN: 978-1-4673-0788-8 978-1-4673-0786-4 978-1-4673-0787-1. DOI: 10.1109/ICQR2MSE.2012.6246300. [Online]. Available: http://ieeexplore.ieee.org/document/6246300/ (visited on 10/16/2021).

[34] T. Devries, K. Biswaranjan, and G. W. Taylor, "Multi-task learning of facial landmarks and expressionopenface," in *2014 Canadian conference on computer and robot vision*, IEEE, 2014, pp. 98–103.

[35] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 2012, pp. 2879–2886.

[36] Y. Tang, "Deep learning using linear support vector machines," *arXiv preprint arXiv:1306.0239*, 2013.

[37] I. J. Goodfellow, D. Erhan, P. L. Carrier, *et al.*, "Challenges in representation learning: A report on three machine learning contests," in *Neural Information Processing*, M. Lee, A. Hirose, Z.-G. Hou, and R. M. Kil, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 117–124, ISBN: 978-3-642-42051-1.

[38] J. M. Susskind, A. K. Anderson, and G. E. Hinton, "The toronto face database," *Department of Computer Science, University of Toronto, Toronto, ON, Canada, Tech. Rep*, vol. 3, 2010.

[39] B. Li and D. Lima, "Facial expression recognition via resnet-50," *International Journal of Cognitive Computing in Engineering*, vol. 2, pp. 57–64, 2021, ISSN: 2666-3074. DOI: https://doi.org/10.1016/j.ijcce.2021.02.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666307421000073.

[40] Y. Nan, J. Ju, Q. Hua, H. Zhang, and B. Wang, "A-mobilenet: An approach of facial expression recognition," *Alexandria Engineering Journal*, vol. 61, no. 6, pp. 4435–4444, 2022, ISSN: 1110-0168. DOI: https://doi.org/10.1016/j.aej.2021.09.066. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110016821006682.

[41] E. Barsoum, C. Zhang, C. C. Ferrer, and Z. Zhang, "Training deep networks for facial expression recognition with crowd-sourced label distribution," in *Proceedings of the 18th ACM international conference on multimodal interaction*, 2016, pp. 279–283.

[42] S. Li, W. Deng, and J. Du, "Reliable Crowdsourcing and Deep Locality-Preserving Learning for Expression Recognition in the Wild," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2584–2593. DOI: 10.1109/CVPR.2017.277.

[43] N. Rawal, D. Koert, C. Turan, K. Kersting, J. Peters, and R. Stock-Homburg, "ExGenNet: Learning to Generate Robotic Facial Expression Using Facial Expression Recognition," *Frontiers in Robotics and AI*, vol. 8, 2022, ISSN: 2296-9144. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2021.730317 (visited on 08/18/2023).

[44] M. Bie, Q. Liu, H. Xu, Y. Gao, and X. Che, "FEMFER: Feature enhancement for multi-faces expression recognition in classroom images," en, *Multimedia Tools and Applications*, May 2023, ISSN: 1573-7721. DOI: 10.1007/s11042-023-15808-w.

[Online]. Available: https://doi.org/10.1007/s11042-023-15808-w (visited on 08/18/2023).

[45] A. V. Savchenko, L. V. Savchenko, and I. Makarov, "Classifying Emotions and Engagement in Online Learning Based on a Single Facial Expression Recognition Neural Network," *IEEE Transactions on Affective Computing*, vol. 13, no. 4, pp. 2132–2143, Oct. 2022, Conference Name: IEEE Transactions on Affective Computing, ISSN: 1949-3045. DOI: 10.1109/TAFFC.2022.3188390.

[46] I. J. Goodfellow, D. Erhan, P. L. Carrier, *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*, Springer, 2013, pp. 117–124.

[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[48] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, PMLR, 2019, pp. 6105–6114.

[49] G. Morshed, H. Ujir, and I. Hipiny, "Customer's spontaneous facial expression recognition," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 3, pp. 1436–1445, 2021.

[50] H. Li, N. Wang, X. Ding, X. Yang, and X. Gao, "Adaptively learning facial expression representation via cf labels and distillation," *IEEE Transactions on Image Processing*, vol. 30, pp. 2016–2028, 2021, Publisher: IEEE.

[51] M. Z. Uzun, Y. Çelik, and E. Başaran, "Micro-expression recognition by using CNN features with PSO algorithm and SVM methods," 2022, Publisher: Int Information & Engineering Technology Assoc.

[52] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Image Analysis: 13th Scandinavian Conference, SCIA 2003 Halmstad, Sweden, June 29–July 2, 2003 Proceedings 13*, Springer, 2003, pp. 363–370.

[53] S. A. R. Sekaran, C. P. Lee, and K. M. Lim, "Facial emotion recognition using transfer learning of AlexNet," in *2021 9th International Conference on Information and Communication Technology (ICoICT)*, IEEE, 2021, pp. 170–174.

[54] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size," *arXiv preprint arXiv:1602.07360*, 2016.

[55] A. G. Howard, M. Zhu, B. Chen, *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[56] Y. Li, X. Huang, and G. Zhao, "Joint local and global information learning with single apex frame detection for micro-expression recognition," *IEEE Transactions on Image Processing*, vol. 30, pp. 249–263, 2020, Publisher: IEEE.

[57] S. Liu, Y. Ren, L. Li, X. Sun, Y. Song, and C.-C. Hung, "Micro-expression recognition based on SqueezeNet and C3D," *Multimedia Systems*, vol. 28, no. 6, pp. 2227–2236, 2022, Publisher: Springer.

[58] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[59] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE signal processing letters*, vol. 23, no. 10, pp. 1499–1503, 2016, Publisher: IEEE.

[60] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, IEEE, 2010, pp. 94–101.

[61] Y. Cheng, B. Jiang, and K. Jia, "A Deep Structure for Facial Expression Recognition under Partial Occlusion," in *2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2014, pp. 211–214. DOI: 10.1109/IIH-MSP.2014.59.

[62] R. Li, P. Liu, K. Jia, and Q. Wu, "Facial Expression Recognition under Partial Occlusion Based on Gabor Filter and Gray-Level Cooccurrence Matrix," in *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, 2015, pp. 347–351. DOI: 10.1109/CICN.2015.75.

[63] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[64] D. Poux, B. Allaert, N. Ihaddadene, I. M. Bilasco, C. Djeraba, and M. Bennamoun, "Dynamic Facial Expression Recognition Under Partial Occlusion With Optical Flow Reconstruction," *IEEE Transactions on Image Processing*, vol. 31, pp. 446–457, 2022, Publisher: Institute of Electrical and Electronics Engineers (IEEE). DOI: 10.1109/tip.2021.3129120.

[65] A. Rodrigues, J. Lopes, R. Lopes, and L. Teixeira, "Classification of Facial Expressions Under Partial Occlusion for VR Games," English, *Communications in Computer and Information Science*, vol. 1754 CCIS, pp. 804–819, 2022, ISBN: 9783031232350, ISSN: 1865-0929. DOI: 10.1007/978-3-031-23236-7_55.

[66] S.-S. Liu, Y. Zhang, K.-P. Liu, and Y. Li, "Facial Expression Recognition under Partial Occlusion Based on Gabor Multi-orientation Features Fusion and Local Gabor Binary Pattern Histogram Sequence," in *2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Oct. 2013, pp. 218–222. DOI: 10.1109/IIH-MSP.2013.63.

[67] X. Huang, G. Zhao, W. Zheng, and M. Pietikäinen, "Towards a dynamic expression recognition system under facial occlusion," *Pattern Recognition Letters*, vol. 33,

no. 16, pp. 2181–2191, Dec. 2012, ISSN: 0167-8655. DOI: 10.1016/j.patrec.2012.07.015.

[68] I. Buciu, I. Kotsia, and I. Pitas, "Facial expression analysis under partial occlusion," in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, ISSN: 2379-190X, vol. 5, Mar. 2005, v/453–v/456 Vol. 5. DOI: 10.1109/ICASSP.2005.1416338.

[69] R. Zhi, M. Flierl, Q. Ruan, and W. B. Kleijn, "Graph-Preserving Sparse Nonnegative Matrix Factorization With Application to Facial Expression Recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 1, pp. 38–52, Feb. 2011, Conference Name: IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), ISSN: 1941-0492. DOI: 10.1109/TSMCB.2010.2044788.

[70] D. Mushfieldt, M. Ghaziasgar, and J. Connan, "Robust facial expression recognition in the presence of rotation and partial occlusion," in *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, ser. SAICSIT '13, New York, NY, USA: Association for Computing Machinery, Oct. 2013, pp. 186–193, ISBN: 978-1-4503-2112-9. DOI: 10.1145/2513456.2513493.

[71] B. Yang, W. Jianming, and G. Hattori, "Face Mask Aware Robust Facial Expression Recognition During The Covid-19 Pandemic," in *2021 IEEE International Conference on Image Processing (ICIP)*, ISSN: 2381-8549, Sep. 2021, pp. 240–244. DOI: 10.1109/ICIP42928.2021.9506047.

[72] S. M.d and M. A. Rahiman, "Symbolic Aggregate approXimation-Local Binary Pattern Feature Descriptor Combination for Automatic Facial Expression Recognition," en-US, *Journal of Computer Science*, vol. 15, no. 1, pp. 45–56, Jan. 2019, Publisher: Science Publications, ISSN: 1552-6607. DOI: 10.3844/jcssp.2019.45.56.

[73] E. Barsoum, C. Zhang, C. C. Ferrer, and Z. Zhang, "Training deep networks for facial expression recognition with crowd-sourced label distribution," in *Proceedings*

of the 18th ACM international conference on multimodal interaction, 2016, pp. 279–283.

[74] Y. Wang, Y. Sun, Y. Huang, *et al.*, "Ferv39k: A large-scale multi-scene dataset for facial expression recognition in videos," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 20 922–20 931.

[75] A. Anwar and A. Raychowdhury, *Masked face recognition for secure authentication*, 2020. arXiv: 2008.11104 [cs.CV].

[76] L. Alzubaidi, J. Zhang, A. J. Humaidi, *et al.*, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Journal of big Data*, vol. 8, pp. 1–74, 2021.

[77] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (Adaptive Computation and Machine Learning series). MIT Press, 2016, ISBN: 9780262337373. [Online]. Available: https://books.google.pt/books?id=omivDQAAQBAJ.

[78] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[79] R. P. Lopes, *Convolutional neural networks*, Slides of Intelligent Systems, Accessed: 2024-10-19, 2022. [Online]. Available: https://virtual.ipb.pt/access/content/group/23d10f60-363a-11ed-b44c-fa163e8fa201/slides/9_cnn.pdf.

[80] A. V. Ikechukwu, S. Murali, R. Deepu, and R. Shivamurthy, "ResNet-50 vs VGG-19 vs training from scratch: A comparative analysis of the segmentation and classification of Pneumonia from chest X-ray images," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 375–381, 2021, Publisher: Elsevier.

[81] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[82] Y. Jie, X. Ji, A. Yue, *et al.*, "Combined Multi-Layer Feature Fusion and Edge Detection Method for Distributed Photovoltaic Power Station Identification," *Energies*, vol. 13, p. 6742, Dec. 2020. DOI: 10.3390/en13246742.

[83] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[84] A. H. Mostafa, H. Abdel-Galil, and M. Belal, "Ensemble model-based weighted categorical cross-entropy loss for facial expression recognition," in *2021 Tenth International Conference on Intelligent Computing and Information Systems (ICICIS)*, IEEE, 2021, pp. 165–171.

[85] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.

[86] R. Scoble, *Neural networks: Simulating the human brain in ai*. [Online]. Available: https://www.unaligned.io/p/neural-networks-simulating-human-brain-ai.

[87] X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, no. 2, p. 022 022, Feb. 2019. DOI: 10.1088/1742-6596/1168/2/022022. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1168/2/022022.

[88] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *Advances in neural information processing systems*, vol. 27, 2014.

[89] W. Gonçalves, M. Santos, L. de Brito, *et al.*, "Deephp: A new gastric mucosa histopathology dataset for helicobacter pylori infection diagnosis," *International Journal of Molecular Sciences*, vol. 23, p. 14 581, Nov. 2022. DOI: 10.3390/ijms232314581.