ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

# TRANSFER LEARNING BASED FACIAL EMOTION RECOGNITION AND ACTION UNIT DETECTION

M.Sc. THESIS

Süleyman Engin BAĞLAYİCİ

Department of Computer Engineering

Computer Engineering Programme

JULY 2021

**ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL**

# TRANSFER LEARNING BASED FACIAL EMOTION RECOGNITION AND ACTION UNIT DETECTION

**M.Sc. THESIS**

**Süleyman Engin BAĞLAYİCİ**
**(504171528)**

**Department of Computer Engineering**

**Computer Engineering Programme**

**Thesis Advisor: Prof. Dr. Hatice Köse**

**JULY 2021**

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

# TRANSFER ÖĞRENME TABANLI YÜZ İFADESİNDEN DUYGU TANIMA VE EYLEM BİRİMİ TESPİTİ

**YüKSEK LİSANS TEZİ**

**Süleyman Engin BAĞLAYİCİ**
**(504171528)**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Bilgisayar Mühendisliği Programı**

**Tez Danışmanı: Prof. Dr. Hatice Köse**

**TEMMUZ 2021**

Süleyman Engin BAĞLAYİCİ, a M.Sc. student of ITU Graduate School student ID 504171528, successfully defended the thesis entitled "TRANSFER LEARNING BASED FACIAL EMOTION RECOGNITION AND ACTION UNIT DETECTION", which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

| | | |
|---|---|---|
| **Thesis Advisor :** | **Prof. Dr. Hatice Köse**<br>Istanbul Technical University | ........................ |
| **Jury Members :** | **Prof. Dr. Hazım Kemal Ekenel**<br>Istanbul Technical University | ........................ |
| | **Prof. Dr. Nafiz Arıca**<br>Bahçeşehir University | ........................ |

**Date of Submission : 11 June 2021**
**Date of Defense : 8 July 2021**

***To my family and my friends who supported me all along the way,***

## FOREWORD

This master thesis is written under the supervision of Prof. Dr. Hatice Köse, at Istanbul Technical University, during the period from fall 2020 until spring 2021.

The thesis aims to build facial emotion recognition and action unit detection models that target children.

My engineering journey has started with mechatronics in my bachelor, and continued with computer science in my master studies. Working in these fields has not only affected my career but also my outlook on life.

I want to thank my supervisor, Hatice Köse, for her excellent guidance and support during my master studies. I'd also like to thank my lab mates, Cemal Gürpınar and Pınar Uluer for being more than great help and being such great co-workers.

I am grateful to my family and my friends for always supporting me with their endless love and understanding. Finally, special thanks to my dear brother Arjen Kılıç, who always inspired me about computer science and encouraged me to move forward.

July 2021 Süleyman Engin BAĞLAYİCİ

**TABLE OF CONTENTS**

**Page**

## ABBREVIATIONS

| | |
|---|---|
| **ASC** | : Autism Spectrum Condition |
| **AU** | : Action Unit |
| **CNN** | : Convolutional Neural Network |
| **EEG** | : Elektroensefalografi |
| **FACS** | : Facial Action Coding System |
| **FER** | : Facial Emotion Recognition |
| **GAN** | : Generative Adversarial Network |
| **GUI** | : Graphical User Interface |
| **LDA** | : Linear Discriminant Analysis |
| **LSTM** | : Long Short-Term Memory |
| **RGB** | : Red Green Blue |
| **RNN** | : Recurrent Neural Network |
| **ROI** | : Region of Interest |
| **SGD** | : Stochastic Gradient Descent |
| **SVM** | : Support Vector Machine |
| **TCAE** | : Twin-Cycle Autoencoder |

## LIST OF TABLES

**Page**

## LIST OF FIGURES

**Page**

## TRANSFER LEARNING BASED FACIAL EMOTION RECOGNITION AND ACTION UNIT DETECTION

## SUMMARY

Facial expressions are one of the primary clues of emotional inference, and facial emotion recognition (FER) is a popular method in the field of emotion recognition. In this study, various deep learning models have been developed to recognize emotions from facial expressions in children. The study covers the development of emotion recognition models to be used in an affectively aware humanoid robot designed for use in hospitals during audiometry tests of hearing-impaired children. The primary purpose of this study, which is part of the RoboRehab project, is to develop a system that will recognize the emotions of children interacting with an assistive humanoid robot. On the other hand, the study is expected to contribute to the literature on emotion recognition and action unit (AU) detection in children.

Emotion recognition in children is more complicated than in adults since children's facial muscles are not fully developed and their faces are relatively small; therefore, children's facial expressions are open to different interpretations. Besides, since hearing-impaired children can have difficulty expressing their feelings, solutions to this problem may support these individuals to get into interactions. In the one part of studies on emotion recognition in the literature, face data is considered whole and emotions are recognized directly by processing the whole face. In the other part of the studies, emotion recognition is realized by detecting the action units that encode facial muscles. The Facial Action Coding System (FACS) was developed to separate emotions from facial expressions by categorizing facial muscle movements into action units. Action units are contractions and relaxations of one or more facial muscles. Facial expressions can be interpreted by a single action unit or by a group of action units. The experiments in this thesis cover both emotion recognition and action unit detection.

The facial emotion recognition process includes data collection, data processing, data labeling, model training, and model testing. Data processing and labeling become challenging due to the amount of data required to train deep learning-based models. Therefore, automating the process of processing and labeling large amounts of data can ease the data preparation phase. LabelFace, a facial expression processing tool, was developed to preprocess datasets for the following the studies. LabelFace aims to provide automatic data labeling and automatic image and video processing. LabelFace offers deep learning models that can recognize emotions and identify action units in children to assist the user in the labeling process. On the other hand, these models can be used for any other research purpose.

LabelFace offers a variety of data manipulation methods that users can apply to datasets. Data processing aims to eliminate the differences (light, noise, background, etc.) between data instances. These processing operations include face detection, face alignment, facial landmark detection, data augmentation, and face masking.

In addition to the data collected from children, the open-source adult dataset CK+ and the open-source children datasets CAFE, DDCF and LIRIS datasets are used in the studies. As one of the contributions of this study, adult face datasets are used to enhance models capability to recognize emotions and detect action units in children. The CAFE dataset consists of poses of children aged 2-8 years with various appearances due to their ethnic origins, while the DDCF dataset consists of poses of 40 boys and 40 girls between the ages of 6-16. The LIRIS dataset consists of spontaneous short videos of 12 children aged 6 to 12 with different ethnic origins. Apart from open-source datasets, facial data are collected from 35 hearing-impaired children between 5-9 years. In the data collection sessions, children watched some video content or interacted with a humanoid robot while taking a test. Children's facial expressions are videotaped throughout the experiments. Frames with high levels of emotional intensity were extracted from the video recordings of the study and labeled for use in the testing phase of our emotion models. Data collected and labeled, including open-source datasets, is finally preprocessed using LabelFace.

After the data collection and preparation process, transfer learning-based model training experiments are carried out. Deep learning models require large amounts of data, which makes it difficult to train deep learning models due to the small amount of data collected on children in the literature. A previously trained model for a similar task can be used as a baseline model in transfer learning, and it can be retrained with new data to adapt it to a new task. Besides, these models can only be used as feature extractor models without retraining. As for tasks related to face data, some pre-trained models trained with a large amount of face data are presented in the literature. These models were fine-tuned with a small amount of children data or adapted to emotion recognition and action unit detection tasks in children as feature extractors.

Experiments are divided as transfer learning with a supervised model and transfer learning with a self-supervised model. In the first part of the experiments, a supervised learning model, VGG-Face is used as a baseline model for knowledge transfer. As another contribution of this work, for transfer learning, the fully connected layers of VGG-Face are replaced with convolutional layers, and custom fully connected layers are placed on top of these layers. The extracted features from convolutional layers are then used for training for the fully connected layers. In the unsupervised learning experiments, the self-supervised learning model, Twin Cycle Auto Encoder (TCAE) is used as a baseline model. In these experiments, the TCAE model structure is not modified, the model is directly used as a feature extractor. Experiments were carried out with Python programming language and open-source deep learning libraries Pytorch and Tensorflow.

Since the data distributions in the collected datasets are generally imbalanced, precision, recall, and F1 score are taken into account instead of accuracy as evaluation metrics. For comparison, the action unit detection performance of OpenFace 2.0, the most widely used open-source facial expression processing tool in the literature, was examined. The action unit detection performances of LabelFace and OpenFace are compared by considering both posed and non-posed children expressions. Nine common AUs (AU0, AU1, AU4, AU5, AU9, AU10, AU12, AU15, AU17) were included in the comparison.

First experiments aimed to label positive, negative, and neutral emotions of children and adults. The fully connected layers on the top of VGG-Face are randomly initialized in the first two experiments, then trained and tested with the CAFE and CK+ datasets separately. In the last experiments, the previously trained CK+ model is fine-tuned with the CAFE dataset to examine the new model's performance in both CK+ and CAFE datasets. Using subject-dependent evaluation, while the pure CAFE model achieved 94.81% accuracy on the CAFE dataset, the fine-tuned model achieved 95.28% accuracy. Fine-tuning the CK+ model with the CAFE dataset improved the recognition accuracy of children's emotions.

For training with VGG-Face for children's AU detection model, CAFE is chosen as the primary training set. The model is tested with DDCF and non-posed children dataset with cross-database evaluation and it is observed that the trained model is successful in terms of generalization and reliability. The model performs better on the posed DDCF dataset with an F1 score of %68.1 and an F1 score of %47.1 on the non-posed children dataset. In the comparison tests, as a first step, OpenFace is tested with the DDCF dataset. OpenFace achieved an F1 score of %24.2 in the evaluation of DDCF frames. OpenFace is also tested with a non-posed children dataset to see the tool's generalization performance in the next step. OpenFace achieved an F1 score of %36.2.

The next step was to fine-tune the children's AU detection model with another dataset to observe whether the fine-tuned model generalizes well to the datasets and performs better. Thus, in the final experiment, the CAFE model was fine-tuned with the DDCF dataset to examine the new model's performance on the non-posed children dataset. The fine-tuned model scored an F1 score of %50.7, outpacing the original model. In addition to mentioned experiments, the action unit detection capability of the children's AU detection models was expanded by adding AU3, AU24 and AU27, which are among the AUs encountered in the training sets.

In experiments conducted under unsupervised learning studies, the TCAE model is used with different experimental mechanisms. The original TCAE model was trained on VoxCeleb datasets with vast amounts of data. Therefore, the collected children dataset is combined with the LIRIS dataset to have sufficient data, and the TCAE model was first trained from scratch with this mixture of datasets in an unsupervised manner. Then, both the original TCAE model and the trained TCAE model are used as feature extractors. When the original TCAE model is used to train a linear classifier with the CAFE dataset, an F1 score of 76.0% achieved in the recognition of 5 emotions on the DDCF dataset. The TCAE model trained with mixed children dataset achieved an F1 score of 51.0% with the same setup. The original TCAE model is also fine-tuned with a mix of children's datasets to observe a potential performance improvement, but no performance improvement was observed.

After the emotion recognition experiments, a separate experiment was performed for action unit detection. In the experiment, nine action units (AU1, AU4, AU5, AU6, AU9, AU10, AU12, AU15, AU17) are detected in the DDCF dataset by a linear classifier trained with the CAFE dataset. The original TCAE model is used as a feature extractor without any fine-tuning, and an F1 score of 63.6% is achieved. When the VGG-Face and TCAE models are analyzed in terms of AU detection performance, it is observed that the generalization success of the VGG-Face model was higher. However, the use of transfer learning and adult datasets has been observed to improve emotion recognition and action unit detection performance in children.

**TRANSFER ÖĞRENME TABANLI YÜZ İFADESİNDEN DUYGU TANIMA VE EYLEM BİRİMİ TESPİTİ**


## ÖZET

İnsan davranışının yorumlanması duygulara önemli ölçüde bağlıdır. Duyguları tanıma yeteneği, psikoloji, eğitim, insan-robot etkileşimi, sağlık, eğlence ve insan davranışıyla ilgili diğer alanlarda önemli bir yer tutmaktadır. Yüz ifadeleri, duygusal çıkarımların temel ipuçlarından biridir ve yüz ifadesinden duygu tanıma (FER), duygu tanıma alanında en çok kullanılan yöntemlerden biridir. Bu çalışmada çocuklarda yüz ifadesinden duygu tanımak için çeşitli derin öğrenme modelleri geliştirilmiştir. Çalışma, işitme engelli çocukların odyometri testleri sırasında hastanelerde kullanılmak amacıyla geliştirilen bilişsel olarak duyarlı bir insansı robotta kullanılacak duygu tanıma modellerinin geliştirilmesini kapsamaktadır. RoboRehab projesinin bir parçası olan bu çalışmanın temel amacı, sosyal ve yardımcı bir insansı robot ile etkileşim halinde olan çocukların duygularını tanıyacak bir sistem geliştirmektir. Diğer yandan, çalışmanın çocuklarda duygu tanıma konusunda literatüre katkı sağlaması beklenmektedir.

Çocuklarda duygu tanıma yetişkinlere göre daha zordur, yüz hatları tam olarak gelişmediğinden duygularını ifade ederken yetişkinlere göre farklı yüz ifadeleri kullanabilirler. Bununla birlikte, özellikle işitme engelli çocuklar duygularını göstermekte zorluk yaşayabildiğinden bu probleme yönelik çözümler bu bireylerin etkileşime girebilmesine destek sağlayabilir. Literatürde duygu tanıma konusunda yapılan çalışmalarda, yüz verisi hem bir bütün olarak ele alınmakta, hem de yüz hareketlerini kodlayan eylem birimlerinin tanınması yolu ile duygu tespiti yapılmaktadır. Eylem Birimi Kodlama Sistemi, yüz kaslarının hareketlerini eylem birimlerine kategorize ederek duyguları yüz ifadelerinden ayırmak için geliştirilmiştir. Eylem birimleri, bir veya daha fazla yüz kasının kasılma ve gevşemelerinden oluşur. Yüz ifadesinin tek bir eylem birimi veya bir grup eylem birimi tarafından yorumlanmasını sağlayabilirler. Bu çalışmada hem duygu tanıma hem de eylem birimi tespit etme deneyleri yürütülmüştür.

Yüzden duygu tanıma süreci, veri toplama, veri işleme, veri etiketleme, model eğitimi ve model testlerini içerir. Derin öğrenme tabanlı modellerin eğitimi için gereken veri miktarı nedeniyle veri işleme ve etiketleme zorlayıcı bir süreç haline gelmektedir. Bu nedenle büyük miktarda veriyi düzenlemek ve etiketlemek için sürecin otomatize hale getirilmesi önemlidir. Çalışmanın ilk kısmında, duygu tanıma modellerini eğitmek için toplanan büyük miktarda veriyi çeşitli görüntü işleme operasyonlarından geçirmek ve etiketlemek için bir yüz ifadesi işleme aracı olan LabelFace geliştirilmiştir. LabelFace, açık kaynaklı araçların tam olarak sağlamadığı servislerle topluluğa katkıda bulunmayı

amaçlamaktadır: 1) Manuel veya otomatik veri etiketleme; 2) Manuel veya otomatik görüntü ve video işleme. LabelFace kullanıcıya etiketleme sürecinde yardımcı olmak için çocuklarda duyguları tanıyabilen ve eylem birimlerini tespit edebilen asistan modeller sunar. Araç, bu modellerle video veya resim verisini otomatik olarak etiketler ve kullanıcıların etiketlerdeki değişiklikleri uygulamaları için düzenleme seçenekleri sunar. Öte yandan, duygu tanıma modelleri herhangi başka bir araştırma amacı için kullanılabilir.

Bir görüntü işleme aracı olarak LabelFace, kullanıcıların veri setlerine uygulayabilecekleri çeşitli görüntü işleme yöntemleri sunar. Amaç verisetleri arasındaki farklılıkları (ışık, gürültü, arka plan vb.) ortadan kaldırmaktır. Bu işleme işlemleri, yüz algılama, yüz hizalama, yüz işaret noktası algılama, veri çoğaltma, histogram eşitleme, yüz kırpma ve yüz maskelemeyi içerir. Kullanıcının tüm bu işleme fonksiyonlarını bir veri kümesine ayrı ayrı uygulayabilmesinin yanında, işlemleri doğrudan bir veri kümesine uygulamak ve eğitime hazır veri setleri oluşturmak mümkündür.

Çalışmalarda çocuklardan toplanan verilerin yanı sıra, açık kaynak yetişkin veri seti CK+, ve açık kaynak çocuk verisetleri CAFE, DDCF, ve LIRIS veri setlerinden faydalanıldı. Bu çalışmanın katkılarından biri olarak, modellerin duyguları tanıma ve eylem birimlerini tespit etme yeteneğini geliştirmek için yetişkin veri kümeleri kullanılmıştır. CAFE veri seti etnik kökenleri nedeniyle çeşitli görünümlere sahip 2-8 yaş arası çocuk pozlarından, DDCF veri seti ise, 6-16 yaşları arasındaki 40 erkek ve 40 kız çocuk pozlarından oluşmaktadır. LIRIS veri seti, farklı etnik kökenlere sahip 6 ila 12 yaşları arasındaki 12 çocuğun spontane kısa videolarından oluşmaktadır. Kullanılan açık-kaynak veri setlerinden CK+, CAFE ve DDCF poz verilerek oluşturulmuş verilerden, DISFA ise poz verilmeden oluşturulmuş verilerden oluşmaktadır. Açık-kaynak veri setleri dışında, 5-9 yaş arası 16 kız ve 19 erkek işitme engelli çocuktan yüz verisi toplanmıştır. Veri toplama sırasında 18 çocuktan farklı duyguları tetiklemek için animasyon filmlerden farklı video içerikleri izlemeleri istendi. Çocukların geri kalanı, bir insansı robot ile etkileşim halindeyken bir tablet üzerinde test yapmalarının istendiği bir çalışmaya dahil edildi. Her iki senaryoda da çocukların yüz ifadeleri deney boyunca videoya kaydedildi. Modellerimizin test aşamasında kullanılmak üzere, çalışmanın video kayıtlarından yüksek duygusal yoğunluk seviyelerine sahip kareler çıkarıldı ve etiketlendi. Açık-kaynak veri setleri de dahil olmak üzere toplanan ve etiketlenen veriler son olarak LabelFace aracılığıyla veri ön işleme süreçlerinden geçirildi.

Data toplama ve hazırlama sürecinin ardından (Transfer Learning) Transfer Öğrenme tabanlı model eğitimlerine başlanıldı. Derin öğrenme modelleri fazla miktarda veriye ihtiyaç duymaktadır, ve bu etken literatürde çocuklarda toplanan veri miktarının azlığı nedeniyle derin öğrenme modellerinin eğitimini zorlaştırmaktadır. Transfer öğrenme ile benzer bir görev için önceden eğitilmiş bir model ana model olarak kullanılabilir, ve az miktarda veriyle yeniden eğitilerek yeni bir göreve adapte olması sağlanabilir. Ayrıca bu modeller yeniden eğitilmeden sadece özellik çıkartıcı modeller olarak olarak kullanılabilir. Yüz verisi ile ilgili görevler konusunda ise literatürde yüksek miktarda yüz verisiyle önceden eğitilmiş, yüz verisinden genel özellik çıkarımında uzman olan bazı modeller sunulmuştur. Deneylerde bu modellerden faydalanılmış, bu modeller az miktarda çocuk verisiyle ince ayarlanarak ya da özellik çıkartıcı olarak kullanılarak çocuklarda duygu ve eylem birimi tanıma görevlerine adapte edilmişlerdir.

Model eğitimlerinin yapıldığı deneyler gözetimli (Supervised) model ile transfer öğrenme ve gözetimsiz (Self-supervised) model ile transfer öğrenme altında iki gruba ayrılmıştır. Gözetimli model ile yapılan deneylerde önceden gözetimli öğrenme ile eğitilmiş ana model olarak VGG-Face modeli kullanılmıştır. Gözetimsiz model ile yapılan deneylerde ise gözetimsiz öğrenme ile eğitilmiş TCAE modeli kullanılmıştır. Kullanılan datasetlerindeki veri dağılımları genel olarak dengesiz olduğundan, değerlendirme yöntemi olarak (accuracy) doğruluk yerine, (precision), (recall), ve F1 puanı metrikleri dikkate alınmıştır.

VGG-Face, eğitim verilerinden özellikler çıkaran bir kodlayıcı ağı olarak kullanıldı. Bu çalışmanın katkılarından biri olarak, transfer öğrenme için, VGG-Face'in (fully connected) tam bağlantılı katmanları, (convolutional) evrişimli katmanlarla değiştirildi ve eylem birimi tanıma için tam bağlantılı katmanlara sahip yeni bir üst model yerleştirildi. Çıkarılan özellikler, özel üst katmanlar tarafından eğitim için kullanıldı. TCAE modeli ile yapılan deneylerde ise model yapısı değiştirilmemiş, model doğrudan öznitelik çıkarıcı olarak kullanılmıştır.

Deneyler Python programlama dili ve açık kaynak kodlu derin öğrenme kütüphaneleri olan Pytorch ve Tensorflow ile gerçekleştirildi. Gözetimli öğrenme deneylerinde, kayıp fonksiyonu olarak Cross Entropy fonksiyonu, iyileştirici olarak ise Stochastic Gradient Descent ve Adam fonksiyonları kullanıldı. Bunların dışındaki eğitim parametreleri genel kullanıma benzer şekilde (RELU aktivasyonu, 16 (batch size) parti büyüklüğü, 1x1 (stride) adım, ortalama kare hatası (MSE), öğrenme oranı (0.0001), momentum (0.9)) uygulandı.

Karşılaştırma için, literatürde en çok kullanılan açık kaynak yüz ifadesi işleme aracı olan OpenFace 2.0'ın eylem birimi tespit performansı incelendi. Her iki aracın eylem birimi algılama performansları, çocukların pozlanmış ya da pozlanmamış veri kümeleri ele alınarak karşılaştırıldı. Karşılaştırma için LabelFace ve OpenFace araçlarının ortak olarak etiketleyebildiği 9 eylem birimi (AU0, AU1, AU4, AU5, AU9, AU10, AU12, AU15, AU17) kullanıldı.

İlk deneyler, çocukların ve yetişkinlerin olumlu, olumsuz ve nötr duygularını etiketlemeyi amaçladı. VGG-Face'in üstüne yerleştirilen tamamen bağlı katman parametreleri, ilk iki deneyde rastgele ilklendirilmiş, ardından CAFE ve CK+ veri kümeleriyle ayrı ayrı eğitilmiş ve test edilmiştir. Son deneylerde, daha önce eğitilmiş CK+ modeli, yeni modelin hem CK+ hem de CAFE veri kümelerindeki performansını incelemek için CAFE veri kümesiyle ince ayarlanmıştır. Deneklere-bağlı değerlendirme yöntemi kullanıldığında, sadece CAFE ile eğitilmiş model %94,81 doğruluk elde ederken, önce CK+ ile eğitilip sonra CAFE ile ince ayarlanmış model %95,28 doğruluk elde etmiştir. Yetişkin veri seti ile eğitilen modele çocuk veri seti ince ayar yapılması, modelin çocukların duygularını tanıma başarısını arttırdığı gözlemlenmiştir.

Çocukların VGG-Face ile eylem birimi tanıma modeli eğitimi için birincil veri seti olarak CAFE seçilmiştir. DDCF veri seti ve toplanan çocuk veri seti ile yapılan değerlendirmeler, CAFE modelinin genelleme ve güvenilirlik açısından başarılı olduğunu göstermiştir. Model, DDCF veri seti ile yapılan testlerde %68.1 F1 puanı elde ederek daha fazla başarı göstermiştir; bu, CAFE'nin pozlanmış bir veri kümesi olması nedeniyle şaşırtıcı değildir. Model, gerçek dünyadaki pozlanmamış verilerde ise %47.1 F1 puanına ulaşmaktadır. OpenFace ile yapılan deneylerde ise ilk adım olarak OpenFace eylem birimi tanıma modelleri DDCF veri seti ile test edilmiştir. OpenFace, 175 DDCF veri örneğinin değerlendirilmesinde %24.2 F1 puanı elde etmiştir. Bir sonraki adımda, aracın genelleme performansını görmek için OpenFace pozlanmamış çocuk veri seti ile test edildi. OpenFace, 198 pozlanmamış karenin değerlendirilmesinde %36.2 F1 puanı elde etmiştir.

Bir sonraki adım, CAFE veri seti ile eğitilmiş modelin genelleme performansını artırmak için modelin başka bir veri kümesiyle ince ayarının yapılmasıdır. Böylece, son deneyde, önceden eğitilmiş CAFE modeli, DDCF veri kümesiyle ince ayarlanmış ve pozlanmamış çocuk veri seti ile test edilmiştir. İnce ayarlı model, %50.7 F1 puanı alarak orijinal modeli geride bırakmıştır.

Bahsedilen deneylerin dışında, eğitim setlerinde karşılaşılan AU'lar arasında yer alan AU3, AU24 ve AU27'nin deneylere eklenmesi ile modellerin eylem birimi tespit kabiliyeti genişletilmiştir.

Gözetimsiz öğrenme çalışmaları altında yapılan deneylerde TCAE modeli farklı deney mekanizmalarıyla kullanılmıştır. Orijinal TCAE modeli, çok yüksek miktarda veriye sahip VoxCeleb veri kümeleriyle eğitilmiştir. Bu nedenle, yeterli veriye sahip olmak için farklı çocuk veri kümeleri birleştirilmiş ve TCAE modeli bu veri kümeleri karışımı ile öncelikle sıfırdan denetimsiz bir şekilde eğitilmiştir. Daha sonra hem orijinal TCAE modeli hem de sıfırdan eğitilmiş model çocuk verisinde özellik çıkartıcı olarak kullanılmıştır. Eğitim parametreleri olarak 128 parti büyüklüğü, Adam optimizer, 0.001 öğrenme oranı, 0.9 momentum tercih edilmiştir. Sigmoid ile ikili çapraz entropi kaybı (binary-cross entropy loss), kayıp fonksiyonu olarak kullanılır. CAFE veriseti ile bir lineer sınıflandırıcı eğitiminde kullanılan Orijinal TCAE modeli DDCF veriseti ile yapılan testlerde %76.0 F1 puanı alırken sıfırdan çocuk verisi ile eğitilmiş TCAE modeli %51.0 F1 puanı almıştır. TCAE modeli son olarak potansiyel bir performans gelişimini gözlemlemek için çocuk veri setlerinin karışımı ile ince ayarlanmıştır, ancak herhangi bir performans gelişimi gözlemlenmemiştir.

Duygu tanıma deneylerinden sonra, eylem birimi tespiti için ayrı bir deney gerçekleştirilmiştir. Deneyde, önceki deneylerle benzer şekilde CAFE veriseti ile bir lineer sınıflandırıcı eğitiminde kullanılan TCAE modeli ile, DDCF veri setinde 9 eylem birimi (AU1, AU4, AU5, AU6, AU9, AU10, AU12, AU15, AU17) tespiti gerçekleştirilmiştir. Orijinal TCAE modelinin herhangi bir ince ayar yapılmadan öznitelik çıkarıcı olarak kullanıldığı testlerde %63.6 F1 puanına ulaşılmıştır. AU tespit performansı açısından VGG-Face ve TCAE modelleri incelendiğinde VGG-Face modelinin genelleme başarısının daha yüksek olduğu görülmektedir. Bununla birlikte, transfer öğrenme ile yetişkin verisetlerinin kullanımının çocuklarda duygu ve eylem birimi tanıma performansını arttırdığı gözlemlenmiştir.

# 1. INTRODUCTION

## 1.1 Purpose of Thesis

The primary purpose of the thesis is to build emotion recognition and action unit detection models that target children. For this aim, transfer learning is employed, and LabelFace, a data labeling, and data processing tool, is developed to manage data labeling and data processing steps. The proposed and implemented models and LabelFace is aimed to be used for further research and practical purposes by the community. The studies covered in this thesis are a part of a project called RoboRehab: Assistive Audiology Rehabilitation Robot, supported by TUBITAK under the project number 118E214. With its predecessor study [3], this thesis aims to develop emotional awareness in an affectively aware robot to assist hearing-impaired children during audiology tests in hospitals.

## 1.2 Literature Review

The first studies on face and emotion formed the basis of facial emotion recognition (FER) studies. The first question that came to the mind was, "Are the emotions and the corresponding facial expressions universal and common?". Ekman et al. [11] claimed that some of the basic emotions are universal and common across cultures. They stated that the six basic emotions: happiness, sadness, fear, disgust, anger, and surprise, correspond to particular facial expressions for people regardless of their ethnicities and cultures.

The Facial Action Coding System (FACS) [12] is derived from analyzing facial movements. Facial movements are categorized into facial action units, that defines as relaxation and contraction of facial muscles. Example action units of upper and lower face can be seen in figure 1.1. The facial action units are then associated with the emotions. A single action unit or a group of action units can be interpreted as different emotions. The facial action units and their composition related to emotion categories can be seen in table 1.1. The FACS is also extended for the infants to Baby FACS [2] as infants show slightly different facial characteristics.

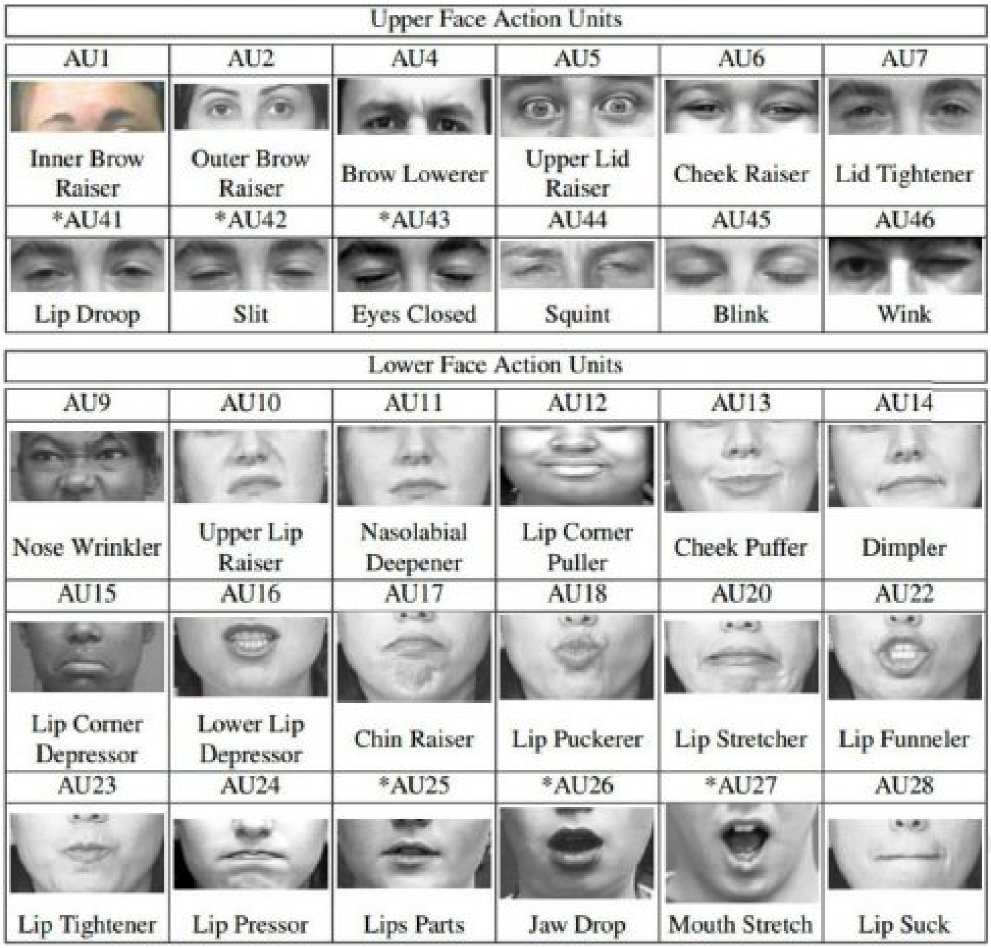

**Figure 1.1 :** Facial Action Units (AUs) of upper and lower face [4]

The development of FACS separated the emotion recognition studies into two bases: Some researchers thought that paying no attention to low-level expressions and directly recognizing an emotion would not be the proper way as the emotions are a composition of these low-level expressions, which are facial action units. Thus, action unit detection seemed as a smaller and more manageable task that eventually leads to emotion recognition. Despite this opinion, many studies have been conducted that try to recognize emotions directly.

From a historical perspective, traditional computer vision methods were the first approaches that researchers used for facial expression analysis. As time goes by, the most significant leaps were achieved with machine learning and deep learning advancements. Having a large amount of high-quality visual data and the proven machine learning methods have enabled researchers to obtain successful

**Table 1.1 :** The facial action units and their compositions related to emotion categories, adapted from [1]

| Category | AUs | Category | AUs |
|---|---|---|---|
| Happy | 12, 25 | Sadly disgusted | 4, 10 |
| Sad | 4, 15 | Fearfully angry | 4, 20, 25 |
| Fearful | 1, 4, 20, 25 | Fearfully surp. | 1, 2, 5, 20, 25 |
| Angry | 4, 7, 24 | Fearfully disg. | 1, 4, 10, 20, 25 |
| Surprised | 1, 2, 25, 26 | Angrily surprised | 4, 25, 26 |
| Disgusted | 9, 10, 17 | Disg. surprised | 1, 2, 5, 10 |
| Happily sad | 4, 6, 12, 25 | Happily fearful | 1, 2, 12, 25, 26 |
| Happily surp. | 1, 2, 12, 25 | Angrily disgusted | 4, 10, 17 |
| Happily disg. | 10, 12, 25 | Awed | 1, 2, 5, 25 |
| Sadly fearful | 1, 4, 15, 25 | Appalled | 2, 9, 10 |
| Sadly angry | 4, 7, 15 | Hatred | 4,7, 10 |
| Sadly surprised | 1, 4, 25, 26 | - | - |

results on emotion recognition. Several studies employ Support Vector Machine (SVM), Random-Forest, and some other machine learning methods to tackle emotion recognition as a classification problem.

Feature extraction is a crucial phase before classification. The aim is to extract useful information from the data that represents the characteristics of the data. Instead of using the data directly, the extracted features are then used for the classification. Geometric-based feature extraction and appearance-based feature extraction are two ways of feature extraction [13], and using one of which or a combination of both is a problem-specific decision.

In recent years, machine learning, more specifically neural networks, got an update with a new paradigm, deep learning. Deep learning utilized and improved the power of neural networks by introducing deep layers of networks. Machine learning methods are highly dependent on the feature extraction and data processing stages. Neural networks with deeper layers provided the advantage of end-to-end training models. That means the elimination of separate data processing and feature extraction phases from the learning process. This paradigm did come with a cost, the cost of the extensive amount of data and the computational power. The amount of data and the computational power required for a deep learning model to interpret and extract features from data is relatively high. The beginning of the big data era and the advancements in the hardware tackled these problems in time.

**Table 1.2 :** Summary of the studies in the literature (C-d: Cross-database evaluation, S-i / S-d : Subject-independent / Subject-dependent evaluation)

| Type | Reference | Data Type | Age Group | Datasets | Approach | Accuracy | F1 score |
|---|---|---|---|---|---|---|---|
| Emotion Recognition | [14] | Static | Adults | JAFFE, CK+ | SRDSAN | 89% | |
| | [15] | Static | Adults | JAFFE, KDEF, FER2013 | Transfer Learning | 43.73% - 92.42% | |
| | [16] | Static | Adults | JAFFE, CK+ | CNN | CK+ 96% JAFFE 98% | |
| | [17] | Static | Adults | CK+, real-subjects | CNN | 28% | |
| | [18] | Dynamic | All | AffectNet, RAF,SFEW, AFEW | CNN + RNN | 52.70% | |
| | [19] | Static | All | MMI, CK+, DISFA, FERA, SFEW, and FER2013 | CNN | C-d 45% S-i 70% | |
| | [20] | Static | All | CAFE, CK+ | CNN | CAFE 66% CK+ 87% | |
| | [21] | Static | All | CK+, JAFFE, Nimstim, MMI | CNN | C-d 48% S-i 77% | |
| | [22] | Static&Dynamic | Adults | JAFFE, MMI | CNN + RNN | 95% | |
| | [23] | Static&Dynamic | Adults | FER2013, TFD, AFEW | CNN + RNN + SVM | 52.80% | |
| | [24] | Static | Adults | CK+, FER2013, NovaEmotions | CNN | C-d 67% - 81% S-i 92%- 98% | |
| | [15] | Static | Adults | FER2013, JAFFE, KDEF, Custom video dataset | Transfer Learning | C-d 81% S-i 65%- 90% | |
| AU Detection | [22] | Static | Adults | BP4D, DISFA | CNN | | C-d 27% S-i 48% |
| | [25] | Static | Adults | BP4D, DISFA, CK+ | CNN | C-d 69% - 96% S-i 76% - 85% | |
| | [26] | Static&Dynamic | Adults | BP4D, GFT | CNN + LSTM | | BF4D 53% - 82% GFT 54% - 66% |
| | [27] | Static&Dynamic | Adults | BP4D | 2D CNN + 3D CNN | | 64.7% |
| | [28] | Static&Dynamic | Adults | DISFA | Transfer Learning | 58% - 74% | |
| | [29] | Static | Adults | FERA2017 | Transfe Learning | 77.8% | 57.8% |
| | [30] | Static | Adults | Custom infant dataset | CNN | 80% | 43% |
| | [31] | Static | Adults | BP4D, DISFA | Transfer Learning | C-d 80.6% S-i 75.2% | C-d 48.5% S-i 55.9% |
| | [32] | Static | Infants | BP4D, DISFA | Multi-task Learning | C-d 92.7% S-i 78.4% | C-d 56% S-i 60% |
| | [33] | Static | Adults | VoxCeleb1, VoxCeleb2, EmotioNet, AffectNet | Self-supervised Learning | EmotioNet 78.6% AffectNet 76.4% | |
| | [10] | Static | Adults | VoxCeleb1, VoxCeleb2, BF4D, DISFA, GFT | Self-supervised Learning | | BF4D 56.1% DISFA 45% GFT 44.2% |
| | [34] | Static | Adults | FERA2017 | Transfer Learning | | S-i 58.2% |

Deep Convolutional Neural Networks (DCNN) is the most used deep learning models for emotion recognition and action unit detection tasks. These networks are applied on not only visual data but also biological signals such as EEG [35]. The success of DCNNs on visual data attracted researchers to include DCNNs in their pipelines. Silvia et al. [17] aimed to find the most successful emotion recognition model to be employed within an affectively aware humanoid robot, Pepper, for human-robot interaction. For this aim, several FER models (Pepper, Google, Microsoft, Kairos, and DCNN) are evaluated and compared in their emotion recognition performance. Angry, happy, neutral, sad, and surprised emotions are considered. The Extended Cohn-Kanade (CK+) [36] database is used for the model training. The models are then tested with the real subjects to assess the performances on real-world conditions. The results showed very distinctive comparisons, but the Pepper model showed the best performance with nearly 60% accuracy on average. The results also showed that recognizing angry, sad, and surprised emotions are more difficult than happy and neutral emotions. In another study, Greco et al. [18] presents a CNN model to be used in Pepper humanoid robot as the emotion recognition module. The model is considered fast enough to run real-time on an embedded device without losing recognition performance. MobileNet and ResNet [37] architectures are implemented with the AffectNet, RAF, SFEW, AFEW datasets. They also use a Temporal Average (TA) method that averages the emotion predictions in a sequence of images. They also integrate Recurrent Neural Network (RNN) to the MobileNet and ResNet to use temporal features in the datasets. Both models are trained from scratch with AffectNet and then fine-tuned with RAF and SFEW datasets. The final models are tested with the AFEW dataset. For convenience, The same evaluation method is applied to similar studies, and the results showed that the ResNet18- TAV method is achieved the best accuracy with 52.7%.

Another study [19] proposes a CNN model supported with Inception Layers to recognize seven basic emotions. The idea of inception layers comes from using more than one filter and pooling operations at the same layer of a neural network, then concatenating the outputs of the operations. This method performs better on region learning, which means extracting local features by choosing the best features among all outputs. Inception layers bring sparsity to the network; thus, the number of layers and parameters will not get larger even though the computational resource needs will be higher on the inception layers. For the experiments, MultiPIE, MMI, CK+, DISFA [38], FERA, SFEW, and FER2013 datasets are used. The evaluations are applied in cross-database and subject-independent manners. The subject-independent evaluations showed around 70% accuracy, and the cross-database evaluations showed around 45% accuracy.

In one study [20], researchers investigate emotion recognition in adults and children and the discriminative aspects between these age groups. Researchers especially focused on the facial landmarks by searching for the most defining features in landmarks and the sufficient number of landmarks for successful emotion recognition. They propose a DCNN model that uses facial landmarks as inputs. They use the Child Affective Facial Expression (CAFÉ) [39] and CK+ datasets for the experiments. The presented results showed that the upper regions of the face are more descriptive than the other regions, making the most effect on emotion recognition performance. The subject-dependent evaluation results were around 66% for the CAFÉ dataset and 87% for the CK+ dataset. Breuer and Kimmel [24] investigates the performance of a CNN model on the detection of action units by setting cross-database experiments. Each model is trained to predict the presence of only one AU instead of predicting for all AUs. The CNN architecture is a standard architecture with convolution, pooling, dropout, and activation layers. The authors also provide the eight action units and their corresponding extracted feature maps overlayed on the subject images. The experiments are carried out with CK+, FER2013, and NovaEmotions datasets. The subject-independent evaluations of each dataset showed an accuracy between 92% and 98%, and the cross-dataset evaluation between datasets showed an accuracy between 67% and 81%.

Rudovic et al. [21] carry out another study targeting the children with autism spectrum. Researchers aim to estimate engagement intensity while communicating with children with Autism Spectrum Condition (ASC), as they have difficulties expressing themselves. For this aim, they perform a set of experiments with a novel CNN model called CultureNet, to see how cultural variations between children or the usage of adult data affects the model performance. The CultureNet model is a fine-tuned version of ResNet architecture, with different in-cultural and out-cultural settings. As a baseline model, ResNet is trained with the AffectNet dataset. The experiments show that using adult data as a baseline is useful but not sufficient, and adding data from the target class to the training significantly improves the model performances. The study highlights the importance of the usage of children's data while targeting children.

Li et al. [40] proposed a custom DCNN that aims to improve the feature extraction process to get better emotion recognition results. The model targeted seven basic emotions. They implemented the Spatial-pyramid pooling (SPP) method similar to the inception-layer approach, except pooling is the only operation that is applied multiple times in a layer. They also apply the cross-connection method to connect feature maps that are the outputs of spatial-pyramid pooling operations. The SSP method is applied at the early stages and the final stages of the network, which is claimed to improve low-high-level features. For the evaluation, CK+, JAFFE, Nimstim, MMI datasets are used. Subject-independent evaluations were around 97%, 69%, 59%, 83%, respectively. For cross-database evaluations, CK+ is used as the training set, and the other datasets are used for testing. The results were low compared to subject-independent evaluation with around 39%, 53%, 52% for JAFFE, Nimstim, and MMI, respectively. Chen et al. [14] built an emotion recognition model for the recognition of 7 basic emotions. They utilize region of interest (ROI) based emotion recognition, which considers only the regions of the upper face (eye region) and lower face (mouth region) to recognize emotions. A deep sparse autoencoder network is used for feature extraction. By using sparsity and autoencoders, researchers aim to increase computational efficiency and feature extraction quality. Sparsity is having some of the weights in a neural network as zeros by penalizing the low weights of neurons since they do not significantly influence the network. On the other hand, autoencoders are subnetworks that are utilized for unsupervised feature extraction. The function of an autoencoder is to produce a dimensionally reduced, distilled representation of the input data. For the classification, the softmax regression method is used. The experiments are conducted with the CK+ and JAFFE datasets, and the SRDSAN model performed slightly better than a CNN model presented in [Convolutional Neural Networks for Facial Expression Recognition with Few Training Samples] on the JAFFE set.

Many studies are carried out for action unit detection and emotion recognition, and CNN models are commonly used among those studies. Zhao et al. [22] utilizes a multi-label learning model that is based on region learning and a CNN model. The model contains a custom layer that is specialized to learn regional representations of the face. The output of a convolutional layer is cropped equally into a piece of feature maps and then fed into the custom layer. The output maps are then concatenated again before feeding the output to the next layer. In the study, action units and their corresponding regions are presented based on the model predictions. The model is trained end-to-end with BP4D [41] dataset and tested with 3-fold cross-validation and subject-independent evaluation. For this evaluatiion, 12 AUs (1, 2, 4, 6, 7, 10, 12, 14, 15, 17, 23, 24) are taken into account. DISFA dataset is also used for cross-database evaluation. Only 9 of the AUs (1, 2, 4, 6, 9, 12, 25, 26) are included in the cross-database evaluation. The same experiment setup is created for the LSVM [42], JPML, AlexNet, ConvNet, LCN and a comparison made accordingly. The DRML model showed the best performance among other models with the F1 score of around 48% for subject-independent evaluation and around 27% for cross-database evaluation.

CNN models lack understanding of temporal features in facial data; thus, their usage is limited to static images. RNN and LSTM models are started to be used to make neural networks more capable since temporal understanding is introduced to the models. Some approaches make use of the RNN and LSTM models individually, and some of them create hybrid networks that utilize CNN and RNN-LSTM models together. Dynamic images (videos, sequence of images) became a suitable data structure to feed into neural networks with that upgrade. Feeding videos directly to a network decreased the data processing steps significantly. One study [43] takes a hybrid approach as they use a cascaded CNN and RNN model to recognize seven basic emotions. CNN model is used a feature extractor that extracts a vector of 200 hidden unit activations through its network. For the training of the CNN model, JAFFE and MMI datasets are used. The extracted features are then passed through the RNN layers.

Finally, a valence value is returned from each RNN node that represents the existence of emotions. The experiments show that using CNN and RNN leads to a significant improvement as the hybrid model shows around 95% accuracy for the subject-dependent evaluation. The authors also compare similar studies on the same datasets and show that the hybrid approach shows competitive results with the state-of-the-art models. A similar study [23] proposes a multi-modal approach with a hybrid CNN+RNN network for the 2015 Emotion Recognition In the Wild (EmotiW) Challenge. In the experiments, TFD, FER2013, AFEW 5.0 datasets are used to recognize seven basic emotions in the short video clips. The audio information from the videos is also used to get audio channel features. The system makes use of three feature layers. Spatial, temporal, and audio features are aggregated with CNN, RNN, and SVM models and then fused with a feature-level fusion process. Feature-level fusion is done by concatenating the output features from each feature layer. Various setups are tested by swapping and merging the training and validation sets, and the models' accuracy is between 44% and 53%.

Another study [25] proposes an end-to-end CNN architecture to tackle the multi-label action unit classification problem. A part of the academy focuses one-label classification on action units because there are X action units in the FACS. Having one model for each action unit seems more convenient and reliable than having one model for all to recognize multiple action units. In contrast, the authors claim that there is shared information between a group of action units on face regions, and it is beneficial to learn these pieces of information by having multi-label learning. For this aim, they include AU1, AU2, AU4, AU5, AU6, AU9, AU12, AU15, AU17, AU20, AU25, and AU26 in their experiments. The proposed CNN architecture consists of 2 convolution stages and two max-pooling stages, followed by two fully connected layers. The softmax function is used for the prediction of class probabilities. CK+, DISFA, BP4D datasets are used for subject-independent evaluation and cross-database evaluation. The input images are normalized, and the face regions are cropped to get standard inputs. The model showed around 76% accuracy on the BP4D dataset and 85% accuracy on the DISFA for the subject-independent evaluation. For the cross-database evaluation, the model trained with BP4D is tested with CK+ and DISFA datasets, and the model that is trained with the DISFA model is tested with BP4D and CK+ datasets. The accuracies of these experiments are between 69% and 96%. They also show the success of the models in predicting the group of action units that share the same expressions by comparing the ground-truth correlations and the predicted correlations.

Chu et al. [26] take a hybrid approach to solve the multi-label AU detection problem. They state that spatial information and temporal information are correlated, and they should be examined together to understand a facial expression better. For this aim, they propose a hybrid architecture where an LSTM model is placed on top of a CNN model. CNN model extracts the spatial features, and the LSTM model sequentially uses these features to get temporal information. The spatial features and temporal features are then fused in a fully connected layer propagated to one last fully connected layer to get 12 AU predictions. For the experiments, videos from GFT and BP4D datasets are preprocessed, and facial landmarks are extracted with IntraFace. The evaluations are made with subject-independent evaluation, which includes 3-fold and 10-fold cross-validation. The authors also compare the feature discrimination performance of HOG, VGG Face Descriptor [44] and their model. They claim that their model is more successful in discriminating the AU-related information than HOG and VGGFace. Researchers also train HOG, CPM, JPML, AlexNet models to compare their model to them based on the detection performance of AU1, AU2, AU4, AU5, AU6, AU9, AU12, AU15, AU17, AU20, AU25, and AU26. The models trained with BP4D showed an F1 score of 82.5% for 10-fold and 53.2% for 3-fold cross-validation. The GFT models showed an F1 score of 66.4% for 10-fold and 54.9% for 3-fold cross-validation. The hybrid models mostly showed better performance than standard CNN models and feature-extractors.

In one study, researchers [27] study on a specialized hybrid approach based on region learning. The authors propose a workflow that includes 3d face registration, patch learning (region learning), spatial learning, and temporal learning. For the 3d face registration, they align the faces with a software called ZFace that uses a cascaded regression model to normalize images. In this way, they consider not only facial expression variations but also head pose variations in data. They determine nine regions from the faces and crop patches from these regions. At the next step, corresponding to each patch, static and dynamic encoders extract spatial and temporal information. For the static encoding, standard 2D CNNs, and for the dynamic encoding, 3D CNNs are used. They also integrate a sigmoidal function into the model to understand how specific patches relate to specific AUs. The model learns some parameters from scratch that indicate the importance of a patch on a particular

AU. Finally, a fully connected layer for each AU is trained individually on the top of the network. The experiments are conducted with the BP4D dataset, and 3-fold subject-independent evaluation is used for the comparison. AU1, AU2, AU4, AU6, AU7, AU10, AU12, AU14, AU15, AU17, AU23, and AU24 are included in the experiments. The D-PattNet model is compared to various state-of-the-art models and outperformed all of them with the F1 score of 64.7% on average. AU1, AU2, and AU23 are seemed to be the hardest AUs to detect, and AU6, AU7, AU12 are detected highly accurately. The authors also state the high co-occurrence of AU6, AU7, AU10, AU12, and AU14 in the facial expressions.

In one crucial study which covers infants [30], researchers investigate action unit detection in infants and carry a comparative study to provide a reliable automatic action unit detector model for infants. They propose an end-to-end multi-label CNN to detect 9 AUs from an extensive dataset that contains thousands of spontaneous videos of infants. 230.000 frames are extracted from these videos and manually coded regarding the Baby FACS [2] by the coders. Frames are then preprocessed with face detection and registration to remove head pose variations between frames. The CNN model, adapted from the AlexNet, has eight layers with convolutional, pooling, and fully connected layers. The researchers verify the reliability of not only the automatic coding but also face registration and manual coding. Different evaluation methods such as positive agreement (F1 score), negative agreement, free margin kappa, and accuracy are used for reliability and validity measures. While the manual coding achieves an F1 score of 51% and an accuracy of 89%, the CNN model achieves an F1 score of 43% and an accuracy of 80%. The subject-independent evaluation showed that AU4 AU9 AU 28 are highly recognized correctly while AU1, AU3, AU12, AU20 are harder to recognize. The results show that the automatic coding of the infant datasets can be reliable and valid as much as manual coding, and further research to improve coder models for this task is promising.

In another FER study [15], researchers set up experiments to assess the suitability of various CNN models for emotion recognition tasks and use transfer learning to measure its effect on the emotion recognition performance. They either train models from scratch or use transfer learning to compare recognition performances. They include different CNN models such as FaceLiveNet, VGG16, InceptionV3, Inception-ResNet V2, and DenseNet201 to the experiments. FER2013, JAFFE, KDEF datasets, and a custom children dataset are used to train and test these models. Unlike FER2013, JAFFE and KDEF datasets contain images that have more aggregated expressions, which lead to better recognition performance on the test data. The FER2013 dataset includes facial expressions with low intensity, making it more suitable to be used as a base model for transfer learning. The experiments show that using a pre-train model for training can significantly improve the recognition performance independent of CNN architecture.

On the other hand, some experiments record a low influence of transfer learning on the performance, proving that the transfer knowledge's effectiveness depends on the cross-dataset characteristics. One study [28] compares pre-trained models regarding their performance on feature extraction and provides different assembled models for action unit recognition. AlexNet, ZFNet, GoogleNet, LeNet-5, VGGNet, and ResNet architectures are included in the experiments. DISFA dataset is used with subject-independent evaluation. Images that contain AUs with an intensity level of 2 and above are used for training and testing. Instead of training one model for all AUs, individual models for each AU are trained. Since each AU has an individual model, they train each model with only AU-related regions cropped from the whole image. Linear Discriminant Analysis (LDA), SVMs, and LSTMs are placed as classifiers on the top of feature extractors. While using an SVM classifier and a fixed feature extractor, ResNet-152 and VGG-Face architectures achieved the best performances with F1 scores of 58.7% and 60.7%, respectively. LSTM classifier performed slightly worse due to high input dimension and low training data size. A significant performance increase is observed when ResNet-152 and VGG-Face architectures are also fine-tuned from a specific layer, and region-based learning is applied. They also ensembled different models using majority voting with various setups (ensembled feature extractors, ensembled classifiers) and observed a slight increase in the performance. Tang et al. [29] worked on fine-tuning individual VGG-Face models for the detection of AU1, AU4, AU6, AU7, AU10, AU12, AU14, AU15, AU17, AU23 separately. Facial Expression Recognition and Analysis (FERA2017) Challenge dataset contains videos of facial expressions with nine different head poses for the experiments. They compare the fine-tuned model with a geometric feature-based baseline model as a benchmark. Based on the subject-independent evaluation, the mean accuracy of the VGG-Face models is 77.8%, which is significantly better than the baseline model with an accuracy of 56.1%. The F1 score of VGG-Face is 57.4%, which is also better than the baseline model with an F1 score of 45.2%. The VGG-Face models specifically performed better on AU6, AU7, AU10, AU12, AU14. The recognition performance on some AUs such as AU17, AU1, AU4, AU23 is comparatively low due to head pose influence on these AUs.

One important study [31] integrates transfer learning with enhancing (attention) layers and cropping layers to build a 3 stage model. Enhancing layers apply a feature extraction regarding the attention maps, which are crafted with facial landmark points. Cropping layers are separate layers trained with cropped patches from the facial expressions for region-based learning. These specially designed layers increase the relatedness of the extracted features. For this aim, a pre-trained VGG19 architecture is separated into five groups of layers, and the 4th and fifth groups are fine-tuned to adapt low-level feature extraction to the target task. BP4D and DISFA datasets are used for the experiments. A balancing mechanism discards the imbalanced distribution of AUs in both datasets before training the models. In the first experiment, EAC-Net is trained and tested with the BP4D dataset in a subject-independent manner. AU1, AU2, AU4, AU6, AU7, AU10, AU12, AU14, AU15, AU17, AU23, AU24 are included in the experiment. The other benchmark models LSVM, JPML, DRML, VGG-Face, E-Net, EAC, are also included for comparison under the same experiment condition, and E-Net and EAC-Net achieved the best results with an F1 score of 52.1%, 55.9%, and accuracy of 74.5%, 75.2%, respectively. In the following experiment, researchers did not consider using DISFA for training because of the low number of subjects and low AU occurrence rates in the dataset. Therefore, the fine-tuned models from the previous experiment are used for testing the DISFA with cross-database evaluation. AU1, AU2, AU4, AU6, AU9, AU12, AU25, AU26 are included in the experiment. E-Net and EAC-Net models achieved the best results again with F1 scores of 44.4%, 48.5%, and accuracy of 75.7% 80.6%, respectively. The study also covers experiments to evaluate the AU detection performance of the fine-tuned models on facial expressions with partial occlusions, and large head pose variations. The models show high reliability against these problems and remove the necessity of preprocessing of images. Albiero et al. [34] take another region-based learning approach with a dynamic patch selection mechanism. The multi-label action unit detection model includes the facial landmark detection model for dynamic region learning and the VGG-Face model for transfer learning. FERA 2017 annotated with AU1, AU4, AU6 AU7, AU10, AU12, AU14, AU15, AU17, AU23 is used for the experiments. Since each AU corresponds to a specific facial landmark group, the patches of each frame can be arranged dynamically by centering the patches based on facial landmarks. Twenty patches are cropped from the extracted features and fed into fully connected layers, then concatenated and fed into classification layers. Dynamic region learning makes the model robust for head pose variations, and the classification performance on different head poses becomes similar. Using dynamic region learning increases the performance of VGG-Face by 6.79%. AU1, AU4, AU15, AU17, AU23 are the AUs that are found hard to detect, while AU6, AU7, AU10, AU12, AU14 detected with a relatively high percentage.

Shao et al. [32] propose a multi-task learning model called JAA-Net, that takes into face alignment and action unit detection tasks together. The relatedness of these tasks makes them suitable for multi-task learning. On the other hand, they integrate hierarchical and multi-scale region learning to extract features from faces independent from the scale of AUs, which leads to a better grasp of AUs with different scales. The authors also propose an adaptive attention mechanism to understand the correlations between face regions and AUs. Global feature extractor is the last module included in the model. The face alignment module outputs facial landmarks, which will be used for attention mapping. Local features are extracted more precisely after attention mapping. The global and local features are then concatenated and finally fed into the AU detection layers. BP4D and DISFA datasets are used for the experiments. In the first experiment, AU1, AU2, AU4, AU6, AU7, AU10, AU12, AU14, AU15, AU17, AU23, AU24 are detected in the BP4D dataset with subject-independent evaluation. AU1, AU2, AU4, AU6, AU9, AU12, AU25, AU26 are detected in the DISFA dataset with cross-database evaluation. JAA-NET is compared with state-of-the-art models like EAC-NET, DRML, JPML and showed a significant performance improvement on subject-independent evaluation with an F1 score of 60% and accuracy of 78.4% and on cross-database evaluation with F1 score of 56% and accuracy of 92.7%.

Some of the studies aim to create highly generalized extensive models for facial attribute learning tasks. These models require large amounts of data, and having that amount of labeled data is generally not the case. An essential study [33] proposes an unsupervised learning model, Facial Attributes-Net (Fab-Net), that learns from thousands of videos with different facial expressions movements. The network is trained to extract joint information about facial landmarks, head pose, and facial expression. To generalize a model to that task, source and target image pairs that convey facial movement is encoded and fed into a decoder network that generates an embedding of the source to the target frame. The embedding is then used to generate an image that should be similar to the target image. The pixel-wise difference between the target image and generated image is forced to be minimized, which optimizes the decoder network weights to extract accurate embeddings. There is also a multiple-source to target case in which each source frame is weighted to have different effects on the target image. In this case, the decoder also learns the attention maps of source frames to assign corresponding weights to source frames. The network is trained with VoxCeleb datasets [45] using subject-independent evaluation. Afterward, the model is tested with MAFL, 300-W datasets for landmark detection; AFLW dataset for pose estimation; AffectNet and EmotioNet for expression recognition. The network is compared with state-of-the-art self-supervised and supervised feature extractors for benchmarking. The results show that Fab-Net performs highly better than the state-of-the-art self-supervised models and shows competitive results with state-of-the-art supervised models. Expression recognition Area Under Curve (AUC) performance is given as 78.6% and 76.4% for EmotioNet and AffectNet, respectively.

Similarly, Li et al. [10] present a self-supervised learning model called Twin-cycle Autoencoder (TCAE) for automatic labeling of extensive datasets. The model captures facial action unit variances and head pose variances between source frame and target frame then separates these variances for feature disentanglement. Using source image and disentangled features, pose-changed and AU-changed frames are generated and fused to being compared with the target image. Moreover, two cycle mechanisms transform generated images back to the source image. In general, the pixel-level and feature-level differences between source, generated, and target images are constantly compared to measure transformation error, and minimizing that error leads to better feature extraction performance. TCAE model consists of a backbone network for standard feature extraction and separate encoder and decoder networks for AU-related and pose-related feature extraction. Similar to Fab-Net, VoxCeleb1 and VoxCeleb2 datasets are used to train the whole network. The model is evaluated on BP4D, GFT, and DISFA datasets, including state-of-the-art supervised and self-supervised models afterward. The evaluations show that TCAE shows competitive results with supervised benchmarks and achieves significantly better than the other self-supervised methods, including Fab-Net. The F1 scores on the BP4D, DISFA, and GFT datasets are 56.1%, 45.0%, and 44.2%, respectively.

Most of the studies mentioned above focus on emotion recognition and AU detection and data collection, annotation, and processing. Data preparation is a crucial part of this field, and there are various proposed methods and approaches for the preparation process. Assuming that the data collection step is finished, annotation of the collected data is the first problem for the supervised learning methods. Manual annotation of large datasets is exhausting; thus, various open-licensed or commercial facial expression analysis tools are proposed to automatically annotate images or videos (a survey on the open-licensed software can be found in [46]). Some of the commercial tools provide machine-learning solutions for emotion and expression recognition such as FaceReader, SkyBiometry, Kairos, Findface, Noldus. Among those, FaceReader provides an emotion recognition system specialized for infants for the first time. OpenFace 2.0 [46] is the most assertive facial expression analyzing tool that provides facial landmark detection, AU detection, head pose, and eye-gaze estimation functions among the open-licensed software. OpenFace 2.0 includes individual SVM models for the detection of the existence and intensity values of each AU. SVM models specially trained for each AU can yield successful results as well as deep learning models. Therefore, OpenFace's automatic annotation feature works well on adults. In the experiments, the capabilities of the OpenFace and our tool LabelFace are compared, in terms of data processing and data annotation on children datasets.

### 1.3 Hypothesis

The aim of this thesis is to develop and implement emotion recognition and AU detection models for children, to support and contribute to child-robot interaction studies. Based on this idea, several experiments are carried out to develop emotion recognition and AU detection models. We collected data from children, and used open-source facial expression datasets, including adults and children data to support our models with sufficient data. In the experiments, only deep learning, especially transfer learning based methods are used. Regarding the conducted experiments, the following hypothesizes are proposed:

Hypothesis 1 (H1): A model pre-trained with an adult dataset for a face recognition task can be used as a baseline model for emotion recognition or AU detection in children.

Hypothesis 2 (H2): Using a pre-trained model that is used for a similar task and fine-tuning the model for the target task can lead to better results instead of training a model from scratch. Thus, fine-tuning a child model with additional datasets will increase the generalization performance of the model.

Hypothesis 3 (H3): A model trained with a mixture of data from people of all age scales will yield lower success rate than a model tuned with data from the targeted age group. Therefore, adults and children should be considered in different contexts in terms of facial expressions.

# 2. METHODOLOGIES

## 2.1 Neural Networks

Neural networks comprise many neurons that hold some numeric values to represent and process data. The neurons are built in a layer shape and connected layer by layer. As in the brain analogy, neurons encode information and transfer it with the synapses that connect neurons. Artificial neurons have similar functionality that lets us build artificial neural networks for numerous problems. In the human brain, electrical input signals travel through the axons. These electrical signals are transformed into chemical signals via synapses and transported to the dendrites that are the door to the neurons. In artificial neuron communication, a value sent by a neuron is multiplied by a weight that represents the synapse factor. This factor represents the effect of one neuron on another neuron. The multiplier value is then transmitted to the respected neuron.

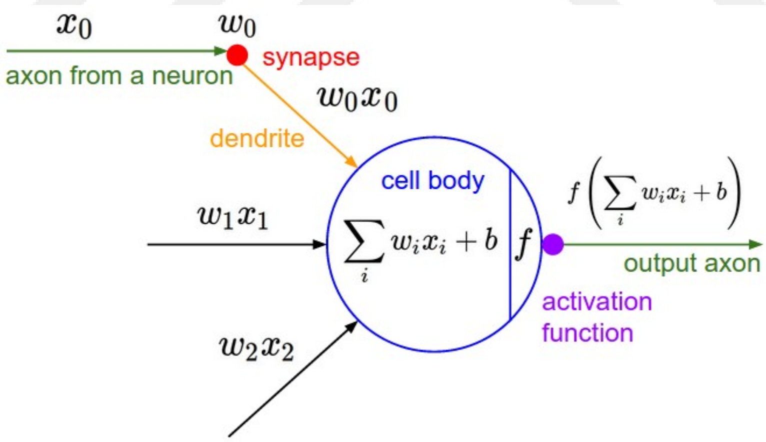


**Figure 2.1 :** Artificial Neuron [5]

With the explained mechanism, each neuron gets input signals from several neurons from the previous input layer, then these input signals are summed, and a final value is used to determine the next value sent by the respected neuron. Again in the brain analogy, neurons get fired if the input signal is above some determined threshold. In artificial neurons, what will be sent to the next neuron is decided by an activation function. These functions take an input value and output another value according to their characteristics. The most-used activation functions are ReLU, Sigmoid, and

Tanh. Activation functions are what give the neural networks the capacity to perform non-linear operations. Having the non-linearity made neural networks superior to the other machine learning models with solving non-linear real-world problems.

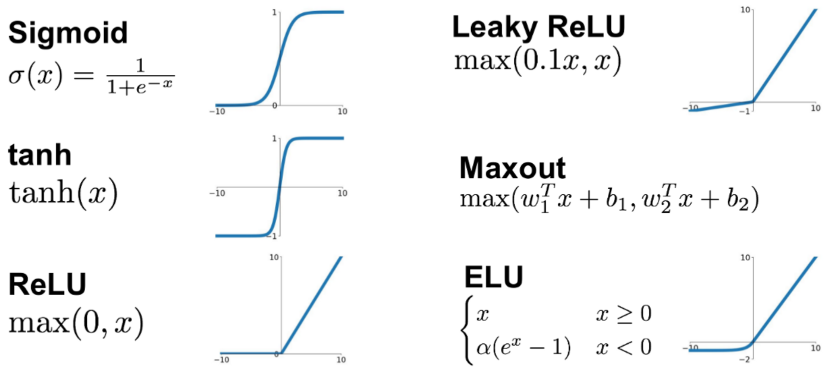


**Figure 2.2 :** Activation functions [6]

Without non-linear activation functions, neural networks can also be used as linear classifiers. A single neuron can return a binary number, which represents a binary classification decision. Putting a loss function to the output of a one-layer neural network and training the network concerning the loss function will create a linear classifier. Training neural networks as a linear classifier on top of a deep neural network was an essential aspect of this thesis and explained in the experiments section in detail.

The learning process is adjusting the values of parameters (weights, biases) of neurons in a neural network. This process goes by predicting the label of incoming training data and tweaking the network parameters according to the accuracy of the predictions. In the beginning, the parameters in the network are initialized randomly, and as the network sees more and more training data, the parameters will be more accurate, and the network will be more successful at predicting. The loss functions determine the measure of how accurate the predictions of the network are. There are different loss functions, but the general idea is a cost function that estimates the difference between the predicted value and the actual value of a training sample. Therefore, the objective is to find the parameters that minimize the cost function output. Minimizing a function with n dimensions means finding the global minimum point of that function in the n-dimensional space. As we know from multi-variable calculus, the gradient of a function shows the direction that decreases the value of the function most quickly. Therefore, the loss function will be minimized by computing the function's gradient, adjusting the parameters as gradient direction, and repeating this process. Gradient Descent is the process of repeatedly nudging the parameters of a function to find a local minimum point based on the gradient direction. To calculate the gradient, one could expect to use the entire dataset, but that is computationally infeasible to show every training sample to the model before computing a gradient. Instead, a random set of samples from the dataset is selected for every iteration of the gradient calculation. That is called Stochastic Gradient Descent (SGD).

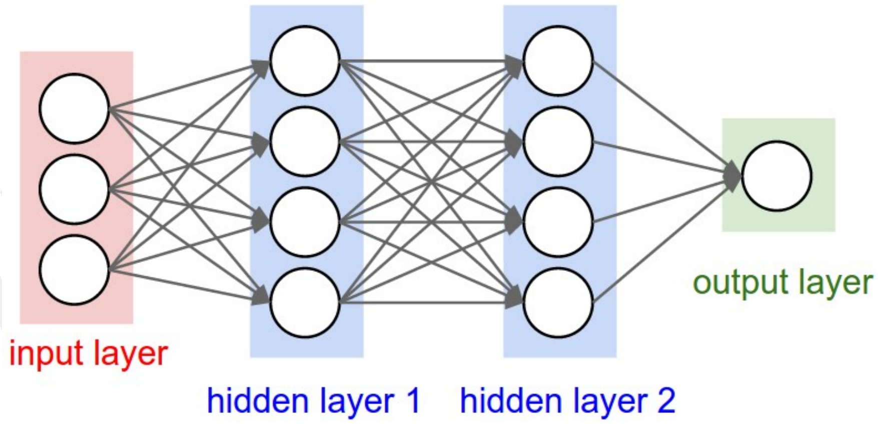

**Figure 2.3 :** Fully Connected Neural Network [5]

When a training sample is showed to the neural network, each neuron will have some activation value, and they will propagate these values to neurons in the next layer. In the forward propagation phase, after the information passed through the final layer, a loss function will be calculated according to the values returned by the neurons in the final layer. Afterward, the backpropagation algorithm calculates the gradient value concerning each neuron's weight. The backpropagation algorithm is an efficient algorithm that does the gradient calculation by using the chain rule. The chain rule with partial derivative expressions indicates how sensitive the loss function is to a specific weight. Three partial derivatives are included in the chain rule:

• The derivative of final loss concerning the activation value of the previous neuron

• The derivate of the activation value of the previous neuron concerning the input from the previous neuron

• The derivative of the input from the previous neuron concerning the weight of the previous neuron

After the backpropagation algorithm is applied to all weights in the network, the weights are updated according to the results. The algorithm is applied iteratively until the weights converge.

## 2.2 Convolutional Neural Network (CNN)

Neural networks were a breakthrough in the pattern recognition area, but they have some flaws that prevent them from being applied to any domain. First of all, the number of parameters gets larger based on the problem to be solved. Having many parameters makes it impractical for some of the problems, such as computer vision, as the problems in vision generally require real-time performance. Another disadvantage of neural networks is, having to transform any data into the numerical form as neural networks accept only numerical representation as inputs. This restriction may cause a performance decrease on the models since a change in data representation can lead to changes in the characteristics of the data. In other words, the models can capture wrong patterns from different representations of the data. Last but not least, the explainability of a network is crucial in some domains, and traditional neural networks are hard to interpret with their structure.

Convolutional Neural Networks (CNN) are proposed against the disadvantages of the traditional neural networks, especially for the computer vision field. CNNs implement the idea of image filtering into the neural networks. CNNs consist of cascaded image filters that have pixel-wise neurons. The idea is to apply different filters on an image in a pipeline manner to extract high-level or low-level features from it. Filters are generally small (i.e., 3x3, 4x4), but they are applied to the whole input image by traversing the image regions. A neuron in each filter is responsible for a linear dot product with incoming data, just like traditional neural networks. Instead of having fully connected neurons, each neuron connects locally to each other, decreasing the number of parameters to be calculated dramatically. The element-wise multiplication of filters with each image region returns some value, and non-linear activation functions follow these operations. ReLU, Leaky ReLU, and ELU are the most used activation functions in CNNs. Each layer operation creates an activation map that includes some spatial information from the input image. The final layers of the CNNs consist of artificial neurons to compute prediction scores. Like traditional neural networks, a loss function is calculated and optimized with forward and backward propagations.

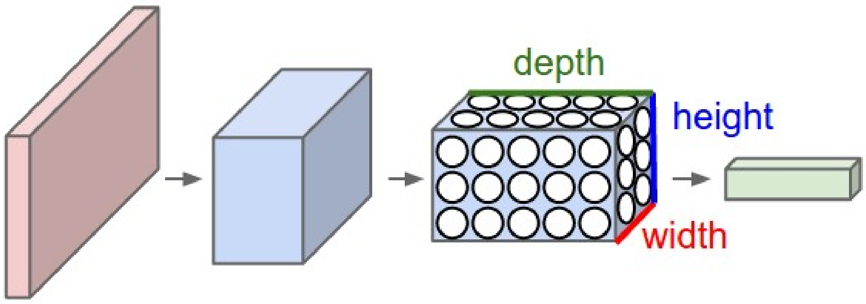


**Figure 2.4 :** Convolutional Neural Network [5]

In addition to convolutional layers, pooling layers apply downsampling operations to the activation maps. This operation may negatively affect the model performance since applying downsampling to the activation maps can eliminate essential features. The important hyperparameters in CNNs are filter dimensions, striding (number of slides of a filter after each convolution operation), and padding (adding artificial pixel boundaries to input activations).

## 2.3 Recurrent Neural Network (RNN)

CNNs do well with static images, as they can extract features from spatial information. However, they are not sufficient to be used in domains where time-dependent information can be requested. Showing an expression or emotion can be seen as a time-dependent process, and understanding this process requires a time-series analysis. Recurrent Neural Networks (RNN) are proposed to detect time-dependent patterns in data. RNNs accept variable-sized input data, which gives them the ability to examine time-dependent input sequences. RNNs include hidden states in their layers. These hidden states keep information from the previous layers that will influence the following layers by participating in the loss and gradient calculation. The vanilla RNN gradient flow can be seen in figure 2.5. As in the figure; X's are the inputs, $h_i$ is hidden state, W's are the weights. At each stage, the input vector is multiplied with its corresponding hidden state variable. The results are then fed into the activation function that outputs the new state variable. RNNs can have any input-output shapes: one-to-one, one-to-many, many-to-one, many-to-many. The basic architecture of many-to-one RNNs can be seen in figure 2.5. In many-to-many cases, each hidden state variable is multiplied with a unique weight matrix to get an output for each layer. Only the last hidden state variable is multiplied with a weight matrix to get single output in many-to-one cases. Many-to-one multi-layer RNN models are highly applied for emotion and action unit recognition using sequential images and videos.

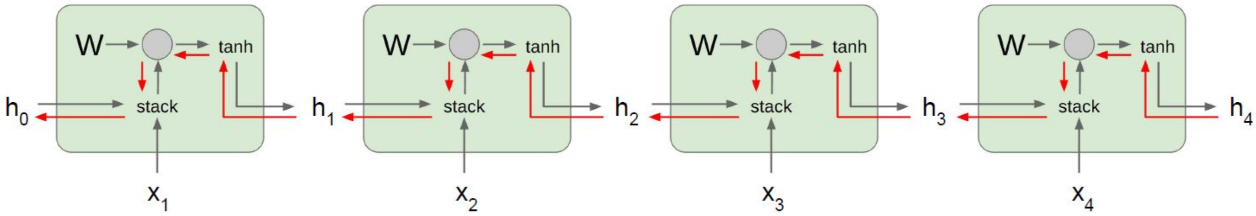


**Figure 2.5 :** Vanilla RNN Gradient Flow [5]

Despite the functionality of RNNs, they can create a problem while training a deep network. The backpropagation algorithm needs to multiply the weight matrix several times for the gradient calculation, and these calculations may result in very low or high values according to weight matrix elements. If the largest element of a weight matrix is above one, several multiplications will make the gradient enormously high, called exploding gradients. On the contrary, if the element is smaller than one, the gradient value shrinks exponentially and becomes very low, called vanishing gradients. Researchers use weight regularization approaches, especially Gradient Clipping, to tackle exploding gradients problems by scaling the gradient based on its L2 norm value. However, the vanishing gradients problem requires a different design.

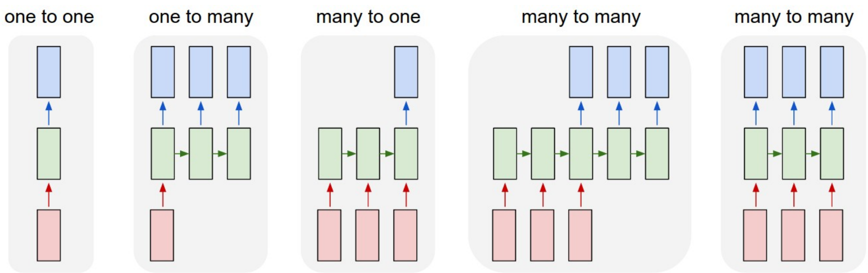


**Figure 2.6 :** RNN architectures [5]

## 2.4 Long Short-Term Memory Neural Network (LSTM-NN)

LSTM networks are designed to alleviate vanishing and exploding gradients. Similar to RNN cells, LSTM cells include hidden state variables to memorize temporal information. Besides, four different gate variables are integrated into an LSTM cell to handle different tasks. Forget gate (f) determines how much we want to use information from the previous cell by eliminating unwanted information. Input gate (i) determines how much do we want to input into our cell. Input modulation gate (g) determines how relevant the inputs are, and it influences the inputs with non-linearity. Finally, output gate (o) determines the output of the cell. The architecture of single LSTM cell can bee seen in figure 2.7.

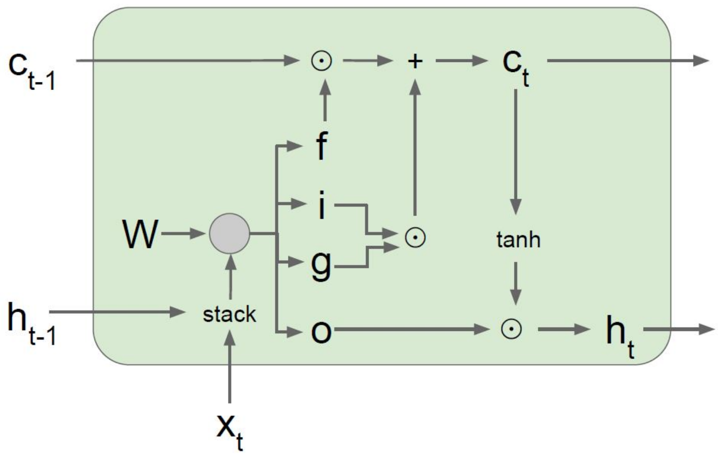


**Figure 2.7 :** Vanilla LSTM cell [5]

The gate variables of an LSTM cell are computed as in the equation 2.1, and the cell state and hidden state are computed as in the equations 2.2 and 2.3, respectively. Each cell takes the previous cell state $c_{t-1}$ and previous hidden state $h_{t-1}$, as well as the current input $X_t$. These variables are then stacked and multiplied with the weight matrix to produce four gates. The forget gate is multiplied element-wise with the cell state. Then the input gate and input modulation gate are multiplied element-wise and then added to the next cell state. The cell state is then activated with tanh and multiplied element-wise with the output gate to produce the next hidden state. İnput gate, forget gate, and output gates use the sigmoid function, and input modulation gate uses tanh function. In the backpropagation step, the gradients of the cell states are easily propagated through the beginning of the network with the upstream gradient flow. The element-wise multiplication with forget gate does not cause a vanishing or exploding gradients problem since forget gate is coming from a sigmoid function (it is between zero and one), and it may vary from cell to cell. The gradient with respect to

W is computed in every cell using cell state and the hidden state, and the local gradients on W at each time step are added together to compute the final gradient on W.

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{pmatrix} W \begin{pmatrix} \mathrm{h}_{\mathrm{t-1}} \\ \mathrm{x}_{\mathrm{t}} \end{pmatrix} \quad \textbf{(2.1)}$$

$$h_{\mathrm{t}} = f \odot c_{\mathrm{t-1}} + i \odot g \quad \textbf{(2.2)}$$

$$c_{\mathrm{t}} = o \odot tanh(c_{\mathrm{t}}) \quad \textbf{(2.3)}$$

## 2.5 Transfer Learning

Learning is dependent on the assumption that the domain of the training set must be similar to the domain of the test set. That means one should have a sufficient amount of data that belongs to some domain to solve a problem in that particular domain, but having a sufficient amount of data is not always possible. In this case, one can exploit the knowledge from related domains by including a model from these related domains in the learning pipeline. Neyshabur et al. [47] present the importance of reusing features from a related domain, and they explain that a model trained from scratch will be in a different feature space than the related domain; thus, the advantage of using a related feature space will be lost. They also show how module criticality is important and which layers are more critical than others. Yosinski et al. [48] investigate the layers of a network regarding the level of features and feature generality. They show that even though transferring knowledge is highly dependent on the similarity between the original domain and the target domain, using a model from a distinct domain would give a better initialization point than random initialization. In transfer learning, instead of training a model from scratch for every problem, a pre-trained model from a related domain can be used as a backbone. These models either can be used as generic feature extractors, or they can be re-trained. To use them as feature extractors, one should remove the fully connected classification layer of these models and use the convolutional layers for getting n-dimensional features. On top of the convolutional layers, different classifiers such as custom fully connected layers, global

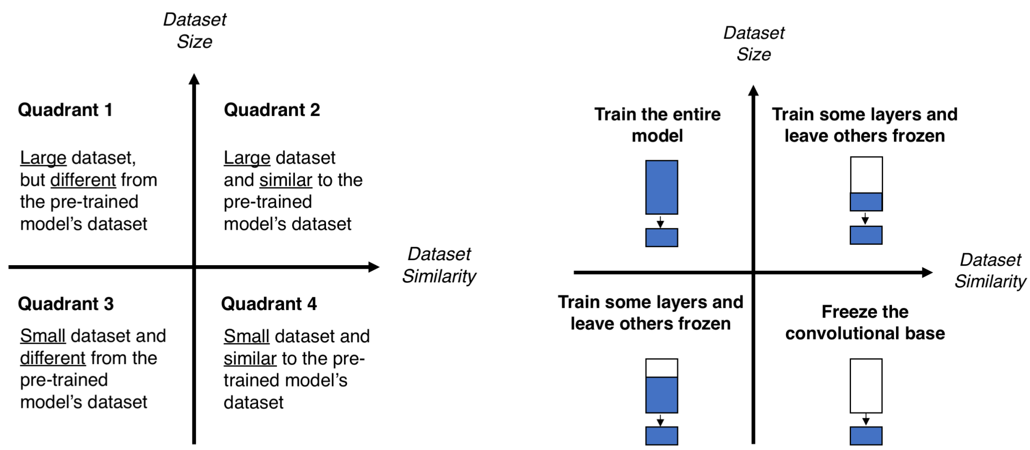


**Figure 2.8 :** Size-Similarity matrix (left) and decision map for fine-tuning pre-trained models (right) [7]

average pooling, or SVM can be placed. Finally, these additional layers can be trained with the new training set to adapt the model to the new domain problem.

Deep learning models capture the high-level patterns (edges, generic shapes) on the shallow layers and low-level domain-specific patterns on the deeper layers. Thus, freezing the shallow layers of a convolutional base and retraining the rest of the model would lead to learning better of the low-level features of the new domain. This approach is called fine-tuning. Deciding on how many layers to retrain is dependent on the size of the dataset and computational power. If the size of the dataset is small, then training many layers may lead to overfitting, or the training may not affect the network at all.

Finally, we can retrain the entire model from the beginning. The new dataset size and dataset variations are the most critical measures to decide between all approaches. If there is a small dataset and less variation between datasets, we can use the pre-trained model as a feature extractor and only train the custom top layers for the classification. If there is a large dataset and less variation between datasets, we can consider all approaches, but the best choice would be fine-tuning a part of the model. If there is a large dataset and high variation between datasets, fine-tuning most of the model or fully training the entire network will be the best approach. Finally, if there is a small dataset and high variation between datasets, the best option would be to find a point to freeze the pre-trained model and fine-tune the rest of the model. For small datasets, using data augmentation methods can contribute to the performance of the model.

The pre-trained feature extractor models are generally benchmark models that are trained with an extensive amount of domain-specific data. In the case of facial emotion recognition, any benchmark model for facial recognition tasks can be considered.

One other important consideration is using a small learning rate while fine-tuning a model. Since the problem domains are related, highly tweaking the model parameters may blur/muddy the model and lead to low recognition performance of the model.

### 2.6 Multi-task Learning

Human perception uses concurrent related task learning, which means using the same sensory information to handle several related tasks. For example, we see the 3d representation of the environment and perform different visual tasks such as depth estimation, segmentation, or object detection. All these tasks use visual information but in a different context. Multi-task learning is emerged to imitate that human behavior with different model architectures. In the context of deep neural networks, various neural network architectures bring together different models that either share layers, features, or parameters to learn multiple tasks together. One way of doing multi-task learning is using the same dataset to learn multiple tasks. That brings the advantage of faster and cost-efficient training. These networks can be built up with shared network layers or shared optimization functions. A baseline network can be used for standard feature extraction, and separate classification or regression layers can be placed on top of that feature extractor. Then, the shared model weights can be optimized by minimizing the loss functions of each task together. On the other hand, a multi-task network can have separate models for each task. In this setting, the joint objective function can be optimized by minimizing the difference between model parameters of individual tasks; hence similar tasks have similar parameters. There are also multi-task networks that use different datasets, especially to benefit from other data types together, such as image, speech, and biological data. Using different datasets is only possible when separate neural network models are used, and they are forced to learn together with loss function optimization. Multi-task learning requires that the considered tasks have similar contexts; thus, learning dissimilar tasks would decrease the model performance. Increasing one model performance can lead to low performance on another task called negative transfer.

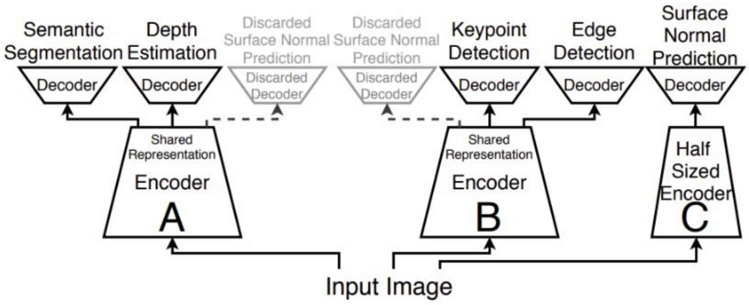


**Figure 2.9 :** An example multi-task learning architecture [8]

Emotion recognition studies involve three different label types: emotions, facial landmarks, and facial action units. These labels correlate to each other; while emotion is a high-level representation, the others are low-level representations of facial expressions. Detecting these labels together with a multi-task learning approach may lead to performance improvement on emotion recognition.

## 2.7 Self-Supervised Learning

Self-supervised learning is an unsupervised learning method that aims to learn representations from unlabeled data. There are already an enormous amount of data we can benefit from on several application domains, and supervised learning limits the usage of that data on these domains because of the necessity of labeled data. Self-supervised learning lets the data supervise the model and shape the model gradually as the model represents the data. Even though the labels are not shown to the model, the model still tries to optimize a function by capturing the relationships between data instances. Self-supervised learning models are generally not expected to be the primary model and give accurate predictions, instead, it is considered a baseline with background information about the predicted task. Therefore, self-supervised models are mostly used as baseline feature extractors for high-dimensional tasks in the literature.

## 2.8 Evaluation

Evaluation methods and metrics are essential to measuring model performances quantitatively and fairly. Most of the studies use cross-database evaluation and subject-independent evaluation methods. In subject-independent evaluation, the subjects in the training set and test set are separated; hence the model will not see a subject in the test set it saw in the training set or vice versa. The application of this method is called K-fold cross-validation, in which the datasets are split into K-folds, then K-1 folds are used to train the model, and the rest is used for testing. This process is carried out for each fold, and the average prediction performance of the model is found to evaluate the generalization performance. Cross-database evaluation is another method that aims to find model generalization performance. In cross-database evaluation, the model is tested with only one dataset in each step, and the average prediction performances on all datasets are calculated similarly to K-fold cross-validation.

| | Positive (Predicted) | Negative (Predicted) |
|---|---|---|
| Positive (Actual) | True Positive | False Negative |
| Negative (Actual) | False Positive | True Negative |

**Figure 2.10 :** Confusion matrix

The most used evaluation metrics in emotion recognition studies are accuracy, precision, recall, and F1 score. Accuracy is the ratio of the number of correct predictions to the number of total predictions. When the number of positive samples, in our case the number of samples with specific emotion or AU in a frame, is not balanced with the number of negative samples in the used datasets, the accuracy metric can mislead to a wrong evaluation.

$$Accuracy(ACC) = TruePositives + TrueNegatives/TotalPredictions \quad \textbf{(2.4)}$$

In the case of imbalanced datasets, precision, recall, and the combination of these, the F1 score gives more accurate estimates. Precision is the ratio of the number of correctly predicted positive samples to the total number of predicted positive samples. The recall is the ratio of the number of samples that are correctly predicted positive to the number of total positive samples. While precision indicates how consistent a model is at predicting the positive samples, recall indicates how successful a model is at finding the positive samples out of the whole dataset.

$$Precision = TruePositives/TotalPredictedPositives \tag{2.5}$$

$$Recall = TruePositives/TotalActualPositives \tag{2.6}$$

F1 score is a function of precision and recall and gives an average performance measure for the datasets with imbalanced class distribution.

$$F1 = 2 * Recall * Precision/Recall + Precision \tag{2.7}$$

# 3. DATA ACQUISITION

## 3.1 Motivation

Data acquisition is the first part of the experiments. For real-world testing, a custom dataset that is collected from a series of Human-Robot Interaction (HRI) experiments is mainly used. On the other hand, well-known adult and children datasets are included in the experiments. Facial expression datasets containing children's data are very few in the literature, and the amount of data in these datasets is not enough to train deep network models from scratch. Thus, even with the data augmentation, they are only handy in the case of model fine-tuning.

## 3.2 Data Collection

The data collection process was carried out as part of a study called RoboRehab [49]. This study aims to use robots with emotion-recognition features to facilitate communication with hearing-impaired children in hospital conditions. For this aim, a custom dataset is collected from 5- to 9- year old, 19 male and 16 female hearing-impaired children. In the data collection experiments, 18 children watched a series of animation videos, and 19 children interacted with a humanoid robot while solving some problems on the tablet. In this way, it was ensured that children show various emotional states while collecting data from them. Most of the known datasets are collected as posed data in the literature, as people are instructed to show a specific expression for each data collection round. Nevertheless, using non-posed data instead of posed data for testing is vital to determine the performance of trained models in real-world conditions.

From the collected videos, static images that contain facial expressions with high-intensity levels are extracted. One hundred ninety-eight frames are obtained from the extraction process.

**Figure 3.1 :** Human-robot interaction study with children [3]

### 3.2.1 Public datasets

*The Child Affective Facial Expression (CAFE)*: The CAFE dataset consists of poses belonging to children of 2 to 8 years of age with diverse ethnic origins and various appearances. The dataset consists of pictures in which children show seven basic emotional states: sadness, happiness, surprise, anger, disgust, fear, and neutral. The dataset has been expanded by children showing the same emotion with different views, either with an open mouth and a closed mouth. The entire dataset consists of 1192 images that show a wide variety of emotions to allow researchers to work on specific problems [39].

*The Dartmouth Database of Children's Faces (DDCF)*: The DDCF dataset consists of pictures taken from 40 boys and 40 girls between the ages of 6 and 16 and containing eight different emotional states. [50] The dataset was created so that images were taken from different camera angles for each emotion under two different light effects. Subjects are dressed in black hats and clothing to reduce the differences between images. As older children show similarities with adults in facial expressions, children aged 6 to 9 were deliberately selected from the DDCF dataset for the trained models for children.

*A novel database of children's spontaneous facial expressions (LIRIS-CSE)*: The dataset consists of spontaneous short videos of 12 children aged 6 to 12 with different ethnic origins. The children expressions contain six basic emotions. The whole dataset includes 208 video clips that are collected in the lab and classroom environments. Video stimuli is used as a trigger mechanism for the expressions. [51]

*The Extended Cohn-Kanade Dataset (CK+)*: CK + is one of the most used adult datasets in the field of FER. The dataset includes the neutral to peak expressions of 18-50-year-old participants, and the peak expressions are fully compatible with the face action coding system (FACS). Five hundred ninety-three image sequences were obtained from a total of 123 subjects. It contains seven basic emotions: anger, respect, disgust, fear, happiness, sadness, and confusion. [36]

### 3.2.2 Data labeling

The datasets mentioned above are used to train emotion assistant models for children and adults. CK+ and DISFA datasets are provided labeled with FACS, but CAFE and DDCF datasets are only available with emotion labels. Thus, CAFE and DDCF are also labeled by manual coders according to FACS and baby FACS. For the adults, the joint AUs in CK+ and DISFA datasets are used. Target AUs for the children and adults can be seen in table 3.1.

**Table 3.1 :** AUs coded by manual FACS coders(* belongs to baby FACS [2])

| AU | Description | Child | Adult | AU | Description | Child | Adult |
|---|---|---|---|---|---|---|---|
| 1 | Inner Brow Raiser | ✓ | ✓ | 15 | Lip Corner Depressor | ✓ | ✓ |
| 2 | Outer Brow Raiser | | ✓ | 16 | Lower Lip Depressor | ✓ | |
| 3* | Brows Drawn Together | ✓ | | 17 | Chin Raiser | ✓ | ✓ |
| 4 | Brow Lowerer | ✓ | ✓ | 20 | Lip Stretcher | | ✓ |
| 5 | Upper Lip Raiser | ✓ | ✓ | 24 | Lip Pressor | ✓ | |
| 6 | Cheek Raiser | | ✓ | 25 | Lips Part | | ✓ |
| 9 | Nose Wrinkler | ✓ | ✓ | 26 | Jaw Drop | | ✓ |
| 10 | Upper Lip Raiser | ✓ | | 27 | Mouth Stretch | ✓ | |
| 12 | Lip Corner Puller | ✓ | ✓ | | | | |

## 3.3 Data Preprocessing

Data preprocessing is a crucial step for most studies in the FER literature. There are different variations between dataset instances or across-datasets such as light conditions, background differences, head pose variations, and many studies have shown that removing these variations results in better performance for the FER models (For a detailed survey about preprocessing methods and the related studies, please refer to [52]). To this aim, some of the processing methods are applied to the used datasets before the experiments.

### 3.3.1 Face recognition

Face detection is the first step of data preprocessing. This process includes detecting the face from an image and cropping the face region to get rid of background information. OpenCV's pre-trained deep face recognition network [53] is used to recognize faces. The model integrates ResNet-10 architecture and Single Shot MultiBox Detector (SSD) model to output bounding boxes that contain a face with high confidence. Also, because SSD is faster than similar models in the literature, it is very suitable for inclusion in the real-time emotion recognition pipeline.

### 3.3.2 Face alignment

Head pose estimation is a subject that has been studied frequently in the literature and is quite determinant in the emotion recognition process. In order to minimize the emotion recognition errors that may occur due to head pose variations, the face alignment method is used. To this aim, dlib's [54] geometric-based facial landmark predictor module is included in the pipeline. For the face alignment, the eye and nose regions of the face are required to be detected. Thus facial landmarks are extracted from faces using dlib's facial landmark prediction module. From the facial landmarks, the central coordinates of each eye and nose are taken. A triangle is made using the nose, eye coordinates, and median of this triangle. Then the center coordinate of the top line of the bounding box containing the face is taken. A second triangle is obtained using the coordinate of the end of the median, the coordinate of the nose point, and the center coordinate of the top line. At the next step, the angle between the first triangle's median line and the long line of the second triangle is found. Finally, to understand in which direction the picture should be rotated, the starting point of the media is rotated around the nose coordinates according to the angle obtained. The picture is rotated with a negative angle if the median point lies in the second triangular space and a positive angle otherwise.

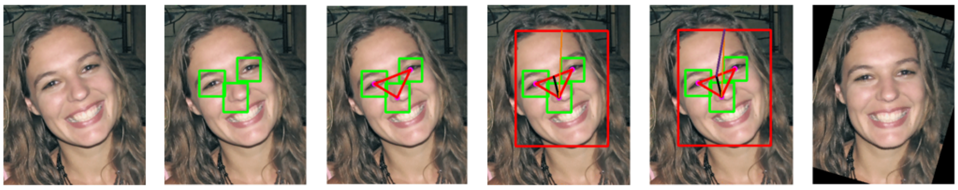

**Figure 3.2 :** Face alignment steps [9]

### 3.3.3 Face masking

Face masking is the last part of image manipulation. It is applied to remove all noise from the data and to keep only the facial data. In this process, facial landmark points are determined from the aligned face first. For facial landmark points in the forehead area of the subjects to fully represent the boundaries of the face, a margin value is added to these points in the 2-dimensional plane. Then, A convex envelope is formed over the boundary points representing the face, and thus the smallest mask covering the face is obtained. The regions outside of this convex set are removed from the picture, and the remaining facial expression is placed on a black background.

**Figure 3.3 :** Original images vs. masked images of subjects belonging to CAFE, DDCF, CK+, DISFA datasets respectively

### 3.3.4 Data augmentation

Since the amount of labeled data required for FER studies is too large, the data augmentation method has been applied in different ways in many FER studies [44]. Recently, some studies investigated the usage of GANs for data augmentation and published promising results [55]. On the other hand, most studies applied geometric augmentation methods on datasets such as translating, rotating, shifting, flipping (reflecting), scaling, cropping, brightening, and darkening. These operations are generally applied at random with different margins. It has been seen that the geometric data augmentation methods are sufficient and adequate for the FER studies; therefore, geometric methods are used in the experiments.

After data processing steps, we obtained 5087 CK+ frames, 2344 CAFE frames, 1190 DDCF frames, and 763 LIRIS frames. Besides, we obtained in total of 31341 video frames, and 198 video frames with high emotional intensity from our collected dataset.

**Table 3.2 :** The distributions of the AUs in the children datasets [3] (* belongs to baby FACS,† are excluded for comparison)

| AU | Description | CAFE (344) | DDCF (180) | Ours (198) |
|---|---|---|---|---|
| 0 | No Action Unit | 71 | 38 | 24 |
| 1 | Inner Brow Raiser | 92 | 47 | 22 |
| 3*† | Brows Drawn Together | 71 | 18 | 16 |
| 4 | Brow Lowerer | 58 | 28 | 7 |
| 5 | Upper Lip Raiser | 61 | 33 | 6 |
| 6 | Cheek Raiser | 51 | 25 | 65 |
| 9 | Nose Wrinkler | 55 | 15 | 10 |
| 10 | Upper Lip Raiser | 55 | 6 | 3 |
| 12 | Lip Corner Puller | 78 | 42 | 88 |
| 15 | Lip Corner Depressor | 41 | 23 | 13 |
| 17 | Chin Raiser | 44 | 39 | 27 |
| 24† | Lip Pressor | 29 | 51 | 54 |
| 27† | Mouth Stretch | 86 | 36 | 6 |

### 3.4 LabelFace: A Facial Expression Processing Tool

Manually performing data labeling and data processing operations is cumbersome, thus automating these processes is essential to focus mainly on model training. Therefore, a data labeling and data processing tool is developed to assist the user with these processes. The tool primarily offers automatic or manual data annotation and various image processing or video processing functions. First of all, the tool is planned to provide the user with ready-to-use emotion recognition and action unit detection models for automatic data labeling. However, a manual labeling option is also provided to label datasets or modify the automatically labeled data manually. LabelFace provides some image processing methods that users can apply to their datasets in terms of image processing. These methods include face detection, face alignment, facial landmark detection, data augmentation, histogram equalization, face cropping, and face masking. In terms of video processing, the tool primarily extracts frames between user-specified time intervals. The user can specify the extraction rate in frames per second (FPS) and the extracted frame size. In addition, features such as video size or resolution changes are included. The tool allows all the features mentioned above to be manually applied to data instances and a pipeline for automatically applying these features to a dataset.

### 3.4.1 Design of the application

The tool's design was carried out with the Qt Designer application, and its development was carried out in the Python programming language with the PyQt library. In its simplest form, the tool is a desktop application where video and audio files can be played and manipulated. The tool uses QtMultimedia and QtMultimediaWidgets to control file playback and playlist management. The interface offers a playlist window where users can control the media files to be played. It is also provided with standard media controls, a timeline scrub widget, and volume control. For the data processing operations, OpenCV, Keras, Scipy, Dlib, and Moviepy libraries are used mainly. One of the features considered in application design is thread-safety. In order for the user to continue using the program while data labeling and data processing operations continue in the background, the thread providing GUI control must be kept separate from the other threads. As it can be seen in figure 3.4, the application interface provides two sections. The first section includes media playing, data labeling, and editing operations, while the second section includes data processing operations.

### 3.4.2 Usage of the application

The data labeling process takes place by the user selecting the time frames from a video and determining the corresponding deep learning model, and then the model automatically labels the frames remaining within this time frame. The user determines the time frame by marking the start time of a facial expression (Frame Start), the peak time of the expression (Frame Peak), and the end of the expression (Frame End). For adult data labeling, action units are offered according to FACS. For children data labeling, some of the AUs from FACS is used, as children do not refer to all AUs

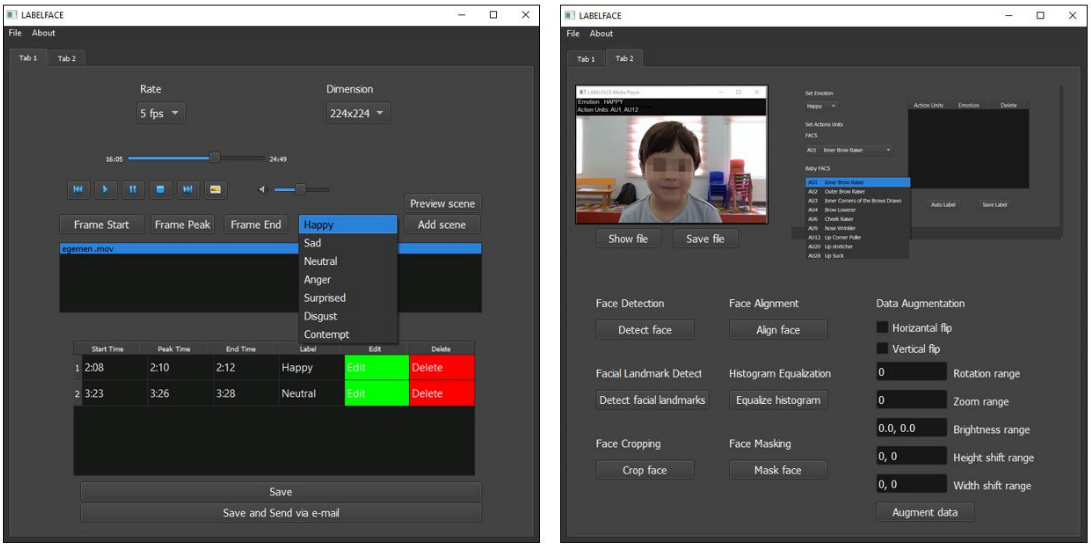


**Figure 3.4 :** LabelFace GUI [3]

presented in the FACS. After picking the specific time frames from the video, the details about the time frames are shown in the bottom widget. The user can either edit the details of a time frame or delete a specific time frame. If the user chooses automatic data labeling and processing, the tool will label the frames between the specified time intervals and then apply the pipeline of processing methods to the frames to get final ready-for-use frames. In our case, these are the aligned and masked images.

# 4. EXPERIMENTS

The whole experiment pipeline that includes data collection, data processing, model training, and emotion&AU prediction can be seen in the figure 4.1. In the first part of the studies, various experiments were carried out with the VGG-Face architecture regarding supervised transfer learning. In the second part of the studies, TCAE architecture is used for unsupervised transfer learning experiments. Pretrained VGG-Face and TCAE models are used for feature extraction, and custom layers are trained with unseen datasets to adapt the system to new tasks.

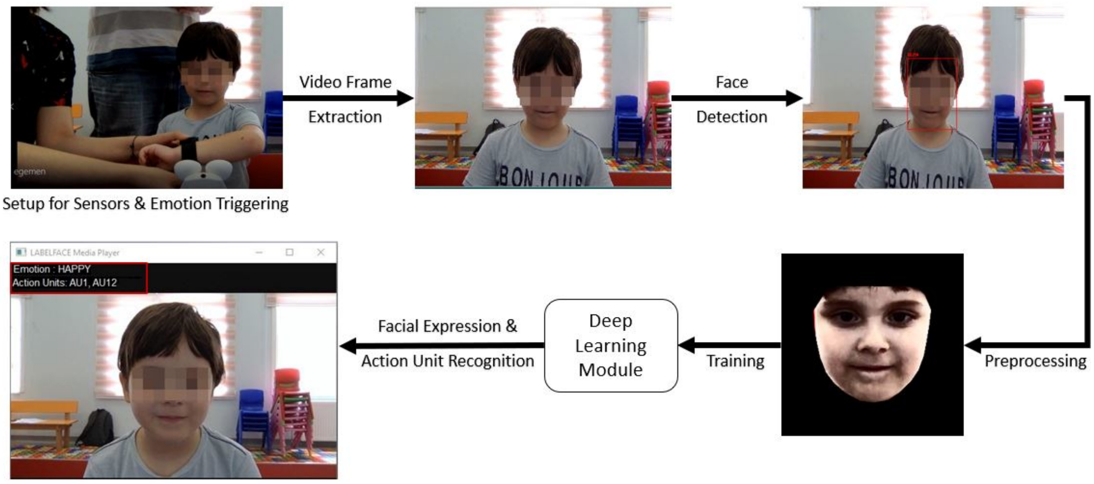


**Figure 4.1 :** Facial expression recognition pipeline [3]

## 4.1 Transfer Learning Using a Supervised Model

VGG-Face is a deep convolutional neural network architecture used for various tasks related to facial recognition. Based on the VGG-Very-Deep-16 CNN architecture [], the VGG-Face has been appropriately modified and used in many studies. The VGG-Face model is trained with 2.6 million facial images, providing a robust model for facial recognition by integrating large amounts of facial data. The model takes 224x224 RGB images as input and performs the classification process using softmax function. In Transfer Learning for Action Unit Recognition, researchers showed that VGG-Face and ResNet are relatively the most suitable pre-trained models for action unit recognition, and the VGG-Net and ResNet variants get the best results. VGG-Face is also one of the most studied architectures in the literature. Based on these motivations and similar features of our dataset derived from real-world setup, VGG-Face is expanded with customized upper layers to meet the requirements of our system best. In the experiments, VGG-Face is used as an encoder network that extracts features from data. For transfer learning, the fully connected VGG-Face layers are replaced with convolutional layers, and a classification layer is placed on top of this architecture for action unit recognition. Features extracted with VGG-Face were used for training the custom top layers. The new base model includes 15 convolutional layers, five pooling layers, and the upper layer containing fully connected layers.


**Figure 4.2 :** VGG-Face architecture [3]

The first tests conducted within the scope of the study aimed to label some of the most used AU's and their corresponding emotions: happy, sad, angry and surprised.Neutral expressions are also labeled without AUs. Besides, a separate experiment is carried out to recognize three emotional states: positive, negative and neutral. The experiments are conducted on several different setups. Only the VGG-Face baseline model kept the same, and different configurations for the top layer are applied. The training parameters are kept the same for fair comparison (RELU activation, batch size (16), steps (1,1), Adam optimizer, learning rate (0.0001), momentum (0.9)). Binary cross entropy loss with sigmoid is used as a loss function. The deep learning models are trained with Python, Keras-Tensorflow libraries.

In the tests where child and adult datasets were used together, it was aimed to label positive, negative, and neutral emotions. For this reason, the emotions of anger and sadness in the CAFE and CK+ datasets were labeled as negative, and the emotions of happiness were used as positive labels. In the first experiment, the top layer has been trained only with the CAFE dataset for 500 epochs and 600 epochs, and the final models are tested with again CAFE dataset. The optimum result has been found at the 500th epoch with 94.81% accuracy. In the second experiment, the top layer has been trained only with the CK+ dataset, and the optimum result has been found at the 200th epoch; thus, training has been stopped at that point. Pure CK+ model showed 92.95% accuracy on testing with CK+ dataset. For the first two experiments, the layers were built in custom and randomly initialized. The next step was fine-tuning a model with unseen data. Thus, in the third experiment, the previously trained CK+ model is fine-tuned with the CAFE dataset to examine the new model's performance in both CK+ and CAFE datasets. The model fine-tuned for 700 epochs, and the best accuracy result for CAFE dataset evaluation is observed at the 600th epoch, which is also surpassed the original CAFE model with 95.28% accuracy. Fine-tuning the CK+ model with CAFE dataset also caused a performance decrease in the test with CK+ dataset. The evaluation of models and the accuracy scores can be seen in table 4.1. Subject-dependent evaluation is used with a 80% & 20% ratio. Setting up the custom layers on the top of a pre-trained face detector model and training two models with CAFE and CK+ datasets separately gives two robust classifiers for both datasets. On the other hand, fine-tuning the CK+ model with the CAFE dataset improved the recognition accuracy of children's emotions. The training history of the three different models can be seen in the figure 4.3.

| Model | Training Set | Test Set | Epoch | Accuracy |
|---|---|---|---|---|
| Pure models | CK+ | CK+ | 200 | 92.95% |
| | CAFE | CAFE | 500 | 94.81% |
| Fine-tuned models | CK+ -> CAFE | CK+ | 600 | 67.60% |
| | CK+ -> CAFE | CAFE | 600 | 95.28% |

**Table 4.1 :** Performance analysis on CK+ and CAFE datasets

The following experiments are carried out to create AU detection models. In order to compare the AU detection performance of the OpenFace models with the models that presented on the child data, 9 common AUs were used. First, OpenFace's AU detection performance on posed child data was tested with the DDCF dataset. OpenFace achieved 18.1% precision, 39.1% recall, and 24.2% F1 score in the evaluation of 175 DDCF frames. In the next step, OpenFace was tested with our non-posed child dataset to test its generalization performance. OpenFace achieved 32% precision, 52.1% recall, and 36.2% F1 score in the evaluation of 198 non-posed frames. The evaluations on OpenFace models showed that the models recall scores was relatively high while precision and F1 scores were lower. This indicates an inconsistency in

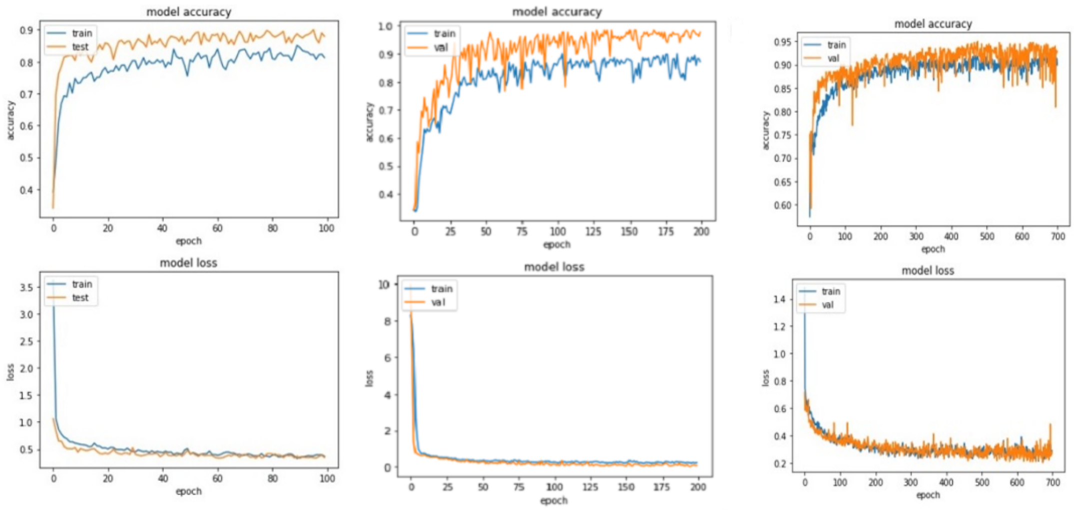


**Figure 4.3 :** Training history: a) CAFE model b) CK+ model c) CK+ fine-tuned with CAFE model

predicting for the true positive samples. For children's AU detection model training, CAFE is chosen as a primary dataset because of its sufficiency in terms of the balance of AU's and relatively large number of frames. The first model trained for comparison with OpenFace was trained with the CAFE dataset and tested with other child datasets. The top layer of the model was trained for 500 epochs, and the best performance was observed around 250 epochs. Evaluations of the trained model with our DDCF dataset and non-posed child dataset show that the CAFE model is more successful on both posed and non-posed datasets. The model achieves an F1 score of 68.1% with the DDCF test and an F1 score of 47.1% on real-world non-posed data. Higher precision scores also indicated the consistent predictions made over the true positive samples.

In the next step, the pre-trained CAFE model was fine-tuned with the DDCF dataset to observe whether there was an increase in AU detection success of a fine-tuned model with another dataset. The model was fine-tuned for 100 epochs and achieved an F1 score of 50.7% in the test with the non-posed child dataset, outstripping the original CAFE model. The precision, recall, and F1 score of evaluations are shown in table 4.2.

**Table 4.2 :** Performance analysis based on 9 AUs

| Training Set | Test Set | Precision | Recall | F1 Score |
|---|---|---|---|---|
| CAFE | DDCF | 75.3 | 66.0 | 68.1 |
| OpenFace 2.0 | DDCF | 18.1 | 39.1 | 24.2 |
| CAFE | Our dataset | 61.1 | 45.6 | 47.1 |
| CAFE + DDCF | Our dataset | 61.9 | 46.5 | 50.7 |
| OpenFace 2.0 | Our dataset | 32.0 | 52.1 | 36.2 |

The evaluations on the DDCF dataset showed that most of the AUs detected successfully while AU9 and AU10 results were comparatively low, and the low number of samples with these AUs in the DDCF dataset makes it difficult to analyze the issue. The AU detection performance on the non-posed children dataset was satisfactory for AU0, AU5 (Upper Lip Raiser), AU9 (Nose Wrinkler), AU10 (Upper Lip Raiser), and AU12 (Lip Corner Puller) while AU1 (Inner Brow Raiser), AU4 (Brow Lowerer), AU15 (Lip Corner Depressor), and AU17 (Chin Raiser) was not properly detected. The uneven distribution of the test set, especially the low number of frames representing some action units, makes it challenging to evaluate AU detection performance. Between the AUs with low success rates, especially AU1 is also found hard to detect in the related studies. Also, the intensity values of these actions highly differ in the non-posed dataset, which may cause the performance difference on the posed and non-posed datasets. One other important finding is that the precision scores of some AUs (AU1, AU4, AU9) are significantly higher than recall scores, and vise versa for some other AUs (AU10, AU15). Higher precision scores indicate that the classifier can be trusted regarding its true positive decisions, but it still can't find the all true positives. On the contrary, a higher recall score and lower precision score indicate an overfitting issue since the classifier is good at finding true positives but it also makes a considerable amount of wrong true positive predictions. The score distributions of 9 AUs are shown in figure 4.4.

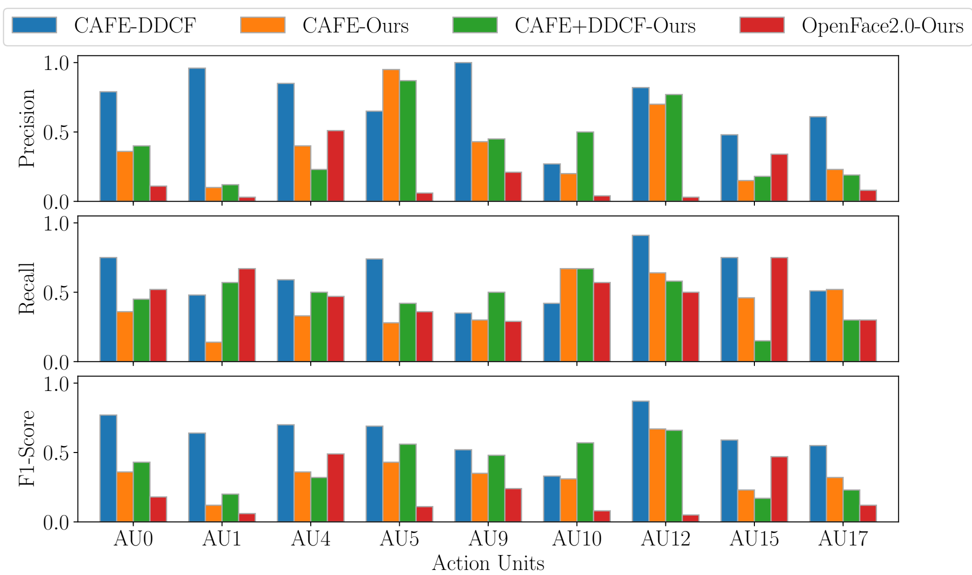


**Figure 4.4 :** Precision, recall and F1 Scores for each AU [3]

In the following study, the AU detection capability of the models was expanded by adding more sample images to the training, which contains AU3, AU24, and AU27. Similar to previous experiments, the top layer CAFE dataset was first trained for 50 epochs and tested with the DDCF dataset. The evaluation of the model with the DDCF dataset over 12 AU shows that the model was successful in terms of generalization and reliability. The model performed better on non-posed data (F1 score 60.5% by DDCF test) and achieved an F1 score of 41% on real-world non-posed data. A pure model was fine-tuned in the next step to observe if there was a performance improvement in a fine-tuned model using a different dataset. To this end, in the final experiment, the pre-trained CAFE model is fine-tuned with the DDCF dataset to examine the performance of the new model with our dataset. The model is fine-tuned throughout the 50 epochs, and the fine-tuned model achieved a better evaluation result with non-posed data, outpacing the original CAFE model with an F1 score of 48%. In the tests performed with the non-posed dataset and the posed dataset, it is observed that the success of AU detection is lower in the non-posed dataset. The factor that causes the performance difference is that the children express their emotions clearly in the posed datasets but naturally take on much more ambiguous facial expressions in the non-posed datasets. In the dataset created with the data obtained from the hearing impaired children during a child-robot interaction game, it is observed that the children showed less arousal in emotions during the interaction with the robot, which reduced the difference between facial expressions of different emotions.

**Table 4.3 :** Performance analysis based on 12 AUs

| Training Set | Test Set | Precision | Recall | F1 Score |
|---|---|---|---|---|
| CAFE | DDCF | 80.6 | 58.4 | 60.5 |
| CAFE | Our dataset | 67.6 | 34.0 | 41.0 |
| CAFE + DDCF | Our dataset | 67.0 | 39.0 | 48.0 |

Precision, recall, and F1 score distributions of 12 AUs are shown in figure 4.5. Most of the AUs showed consistency with the first evaluation with 9 AUs. On the DDCF evaluation, AU1 detection performance is significantly changed where precision score got lower and recall score got higher, which indicates a performance improvement. Among the newly added AUs, AU3 detection performance was promising with CAFE-DDCF evaluation but needs further investigation on the non-posed dataset.

In the AU24 evaluation, an overfitting problem has emerged in the CAFE-DDCF and CAFE-children dataset evaluations, but fine-tuning CAFE with DDCF caused a significant improvement. Finally, AU27 evaluations were fruitful in that they showed consistent success on all evaluations.

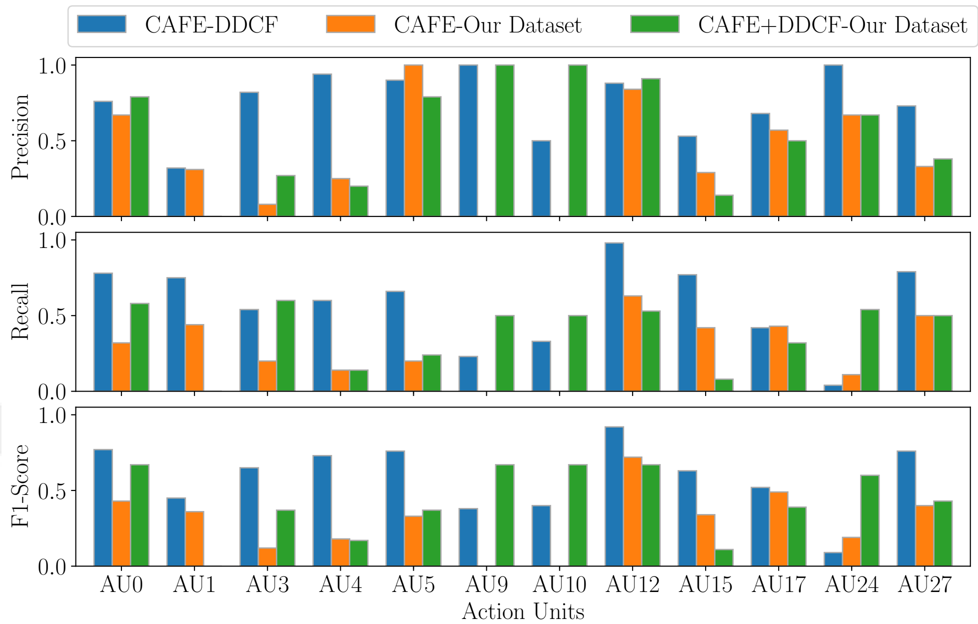


**Figure 4.5 :** Precision, recall and F1 Scores for each AU [3]

The difference in AU recognition performance in children between the LabelFace and OpenFace models depends on several reasons. As the main factor, OpenFace aims to respond to all age scales in AU detection. Therefore, OpenFace AU detection models were trained with datasets containing both child and adult face images. However, in our study with children, a similar success to OpenFace's success in adult data was achieved with children using only child datasets. Therefore, it can be concluded that the detection of AU for children and adults should be considered in different contexts. Another important reason for the performance difference between the LabelFace and OpenFace models is the method differences used for AU detection. It is also evident in the literature that using transfer learning with neural networks instead of a linear SVM classifier gives better results in complex tasks such as AU detection.

## 4.2 Transfer Learning Using a Self-Supervised Model

For unsupervised transfer learning experiments, TCAE architecture is chosen. One crucial motivation in choosing TCAE is that it is built for the automatic labeling of extensive datasets. Thus, training the TCAE model for children can help researchers to label their children's datasets automatically. The original TCAE model is trained with VoxCeleb datasets that have an enormous amount of data. For the experiments, different children datasets are combined to have a sufficient amount of data, and the TCAE network is trained in an unsupervised manner with this mixture of datasets. Merging the custom children dataset and LIRIS dataset, 32104 frames are collected. TCAE model is then used as a feature extractor while training linear classifiers. The training parameters are kept the same for fair comparison (batch size (128), Adam optimizer, learning rate (0.001), momentum (0.9)). Binary cross entropy loss with sigmoid is used as a loss function. The deep learning models are trained with PyTorch library.

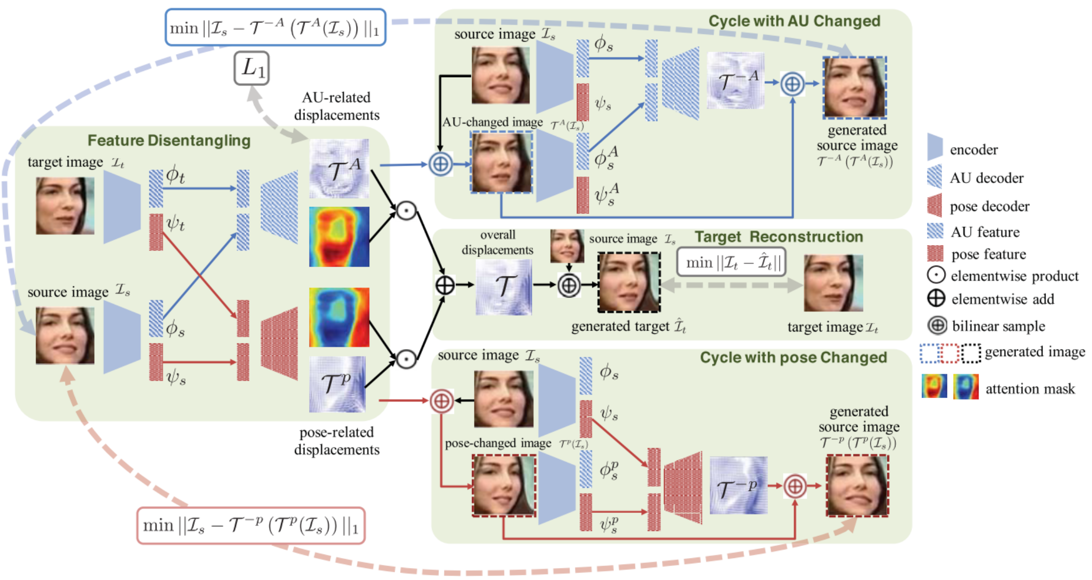


**Figure 4.6 :** TCAE architecture [10]

In the first experiment, the TCAE model is trained from scratch with the mixture of children datasets; then, a linear classifier is trained with the CAFE dataset to recognize five emotions in the DDCF dataset. The optimum result is found at the 1000th epoch, with the mean F1 score of 51%. The recognition performance of five emotions seemed balanced with the following F1 scores: angry: 57%, happy:49.8%, neutral:44.1%, sad: 52.5%, and surprised: 51.5%. In the second experiment, a pretrained TCAE model that is trained with VoxCeleb datasets is used as a feature extractor with and without fine-tuning process. A linear classifier is placed at the top TCAE, and the classifier is trained with the CAFÉ dataset to recognize five emotions in the DDCF dataset. The model showed promising results with the mean F1 score of 76% without any fine-tuning. The TCAE model is also fine-tuned with the mixture of children dataset to observe a potential performance improvement, but improvement is not observed. The result shows that robust feature extractors can be acquired with a sufficient amount of data even if the model is trained fully with adult data. On the other hand, fine-tuning the whole model with a relatively small amount of children data seemed insignificant in the model performance. Increasing the number of training samples or fine-tuning only higher levels of layers may lead to performance improvement.

After emotion recognition experiments, TCAE is used for AU detection experiments. The experiments are carried out to detect 9 AUs (AU1, AU4, AU5, AU6, AU9, AU10, AU12, AU15, AU17) in the DDCF dataset. The pretrained TCAE model that is trained with VoxCeleb datasets is used as a feature extractor without any fine-tuning. In A linear classifier is trained on top of the TCAE using the CAFE dataset to detect 9 AUs on the DDCF dataset, and the mean F1 score of 63.6% is achieved.

**Table 4.4 :** Transfer learning performance analysis of TCAE model based on 5 emotions, and 9 AUs

| Emotion/AU | Pretrained model | Training Set | Test Set | F1 score |
|---|---|---|---|---|
| Emotion | Mixed Children dataset | CAFE | DDCF | 51.0% |
| | VoxCeleb | CAFE | DDCF | 76.0% |
| AU | VoxCeleb | CAFE | DDCF | 63.7% |

Among the evaluated AUs, most of the AUs detected with high success rates except AU10, which is represented very little in the DDCF dataset. Precision, recall, and F1 score distributions of 9 AUs are shown in figure 4.7

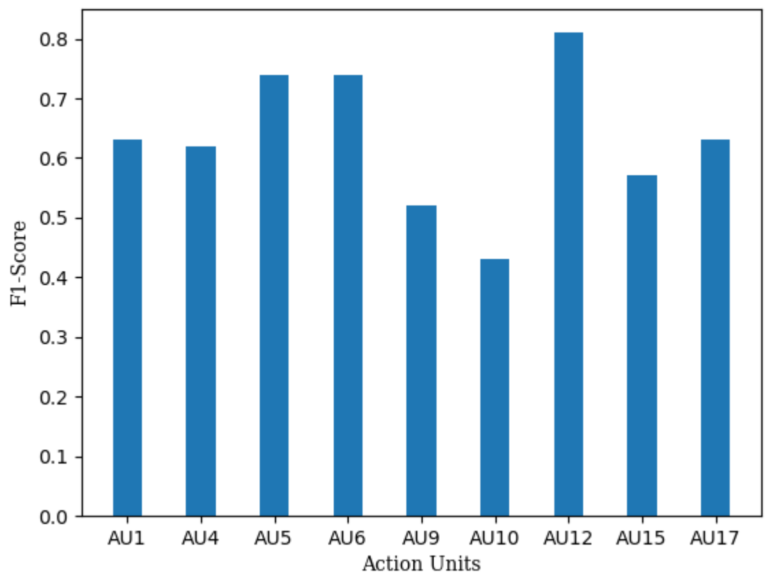

**Figure 4.7 :** F1 Scores for each AU

## 5. CONCLUSIONS AND RECOMMENDATIONS

This study covers the development of emotion recognition and action unit detection models with transfer learning, and the development of LabelFace, a face processing tool that can automatically label data using these models. The system is designed and developed primarily for use in human-robot interaction studies with hearing-impaired children and affectively aware humanoid robots. Recognition of emotion from children's facial expressions is a subject that has not been fully covered in the literature, thus studies in this field have the potential to directly contribute to research related to children. In recent years, significant progress has been made with the use of deep learning models in the field of emotion recognition. Therefore, in this study, the performance of supervised and self-supervised learning models on child data is tested. Due to the fact that the amount of dataset containing child data is insufficient in the literature unlike adults, transfer learning is used in the experiments. With transfer learning, it is aimed that a deep learning model that is previously trained for a similar domain is retrained or used as a feature extractor with new domain data, that way the model will be successful in its targeted new task. Pre-trained models are categorized into supervised learning and self-supervised learning. VGG-Face model, which has been frequently used in the literature for supervised learning studies, is used as the baseline model for one part of the studies. TCAE model, which is the state-of-the-art self-supervised learning model for AU detection is used for the other part of the studies. As far as we know, the proposed children AU detection models are the first to target children aged 5-9 years.

One of the most challenging steps in emotion recognition from facial expression is data labeling and processing. Although many tools are presented to label visual data, the number of tools that can process especially on child data in the field of emotion recognition is few. However, these tools cannot achieve the desired success on child data because they are customized on adult data. In this study, LabelFace, an emotion and action unit labeling, and facial expression processing tool is presented.

LabelFace includes manual and automatic data labeling on datasets, as well as various image processing functions for preprocessing face data. Thanks to the prepared pipeline, the user can automatically prepare the dataset for model training. LabelFace provides emotion recognition and AU detection models to help users label their data regardless of data size. OpenFace 2.0, a face data labeling and processing tool frequently used in the literature, was included in the comparison in order to measure the success of automatic data labeling of models trained with transfer learning. OpenFace 2.0 provides the user with AU detection models used to label AUs, similar to LabelFace, and several experiments have been set up to see the success of these models on children's data. In contrast to the AU-specific models used in OpenFace 2.0, multi-label AU detection models are trained in the experiments. The experiments included two posed child datasets, CAFE, DDCF, and a custom non-posed child dataset collected within the scope of this study. Also a posed adult dataset, CK+ is used.

In the first part of the studies, the fully connected layers of the VGG-Face model are replaced with convolutional layers, and a custom classification layer that contains a number of fully connected layers is placed on top of this architecture. Features extracted from VGG-Face are used to train the custom top layers.

In the experiments where child and adult datasets were used together, it is aimed to label positive, negative, and neutral emotions. Four experiments are conducted using subject-dependent evaluation with an 80% & 20% ratio. In the first and second experiments, the models are randomly initialized, then trained and tested with the CAFE and CK+ datasets separately. In the last experiments, the previously trained CK+ model was fine-tuned with the CAFE dataset to examine the new model's performance in both CK+ and CAFE datasets. While the pure CAFE model achieved 94.81% accuracy on the CAFE dataset, the fine-tuned model achieved 95.28% accuracy. Fine-tuning an adult model with a children dataset improved the recognition accuracy of children's emotions.

Next, the OpenFace AU detection models are compared with our models in terms of detecting 9 AUs on both posed and non-posed child datasets. OpenFace achieved a 24.2% F1 score on posed frames and a 36.2% F1 score on non-posed frames. The model trained for comparison with OpenFace is trained with the CAFE dataset and tested with other child datasets. The model achieved an F1 score of 68.1% on posed data and an F1 score of 47.1% on non-posed data. In the next experiment, the pre-trained CAFE model is fine-tuned with the DDCF dataset and achieved an F1 score of 50.7% in the test with the non-posed child dataset, outstripping the original CAFE model.

The experiments showed that the VGG-Face based multi-label AU detection models have significantly surpassed the OpenFace models on the detection of children AUs. The primary reason for the performance difference is that OpenFace models are trained with data from people of all age ranges to recognize emotions in all age ranges, yielding worse results than a model trained with data only for the targeted age group. Therefore, adults and children should be considered in different contexts in terms of facial expressions. Moreover, OpenFace models are SVM classifiers that are not as good as neural networks in complex tasks such as action unit detection.

An additional experiment was carried out with VGG-Face to expand the AU detection capability of the model by adding AU3, AU24, and AU27 samples to training. The model achieved an F1 score of 60.5% on non-posed data and an F1 score of 41% on real-world non-posed data. The model is then again fine-tuned with DDCF and showed a better evaluation result with non-posed data, outperforming the original CAFE model with an F1 score of 48%. The experiments showed that the success of AU detection performance was lower in the non-posed datasets, caused mainly by the expressiveness degrees between the non-posed and posed datasets.

It is observed that a classifier's performance increases by preprocessing of the datasets, which is also observed in previous studies in the literature. Moreover, placing specially designed layers on top of the pre-trained VGG-Face, and training these layers with a new dataset has adapted well the system to the new task. Also, using an adult dataset at the start of the training, instead of random initialization increased the performance of the classifier. Finally, it is observed that when another dataset with similar characteristics is used to fine-tune a model, the model's generalization performance to another dataset is significantly increased.

In the second part of the experiments, TCAE architecture was chosen for transfer learning with a self-supervised model. First, the LIRIS dataset and custom children dataset are combined to have a sufficient amount of data for the experiment. In the first experiment, the TCAE model is trained from scratch with the mixture of children datasets; then, a linear classifier on top of the TCAE is trained with the CAFE dataset using the trained TCAE model as a feature extractor. The final model recognized five emotions in the DDCF dataset with an F1 score of 51%. In the second experiment, a pretrained TCAE model that is trained with VoxCeleb datasets is used as a feature extractor with and without fine-tuning process. A linear classifier is placed at the top TCAE, and the classifier is trained with the CAFÉ dataset to recognize five emotions in the DDCF dataset. The model showed promising results with the mean F1 score of 76% without any fine-tuning. The TCAE model is also fine-tuned with the mixture of children dataset to observe a potential performance improvement, but fine-tuning did not have any effect.

After emotion recognition experiments, TCAE is used for AU detection experiments. The experiments are carried out to detect 9 AUs (AU1, AU4, AU5, AU6, AU9, AU10, AU12, AU15, AU17) in the DDCF dataset. The pretrained TCAE model that is trained with VoxCeleb datasets is used as a feature extractor without any fine-tuning. The linear classifier is trained on top of the TCAE using the CAFE dataset to detect 9 AUs on the DDCF dataset, and the mean F1 score of 63.6% is achieved.

The result shows that robust feature extractors can be acquired with sufficient data even if the model is trained fully with adult data. On the other hand, fine-tuning the whole model with a relatively small amount of children data seemed insignificant in the model performance. Increasing the number of training samples or fine-tuning only higher levels of layers may lead to performance improvement.

When the VGG-Face and TCAE models are compared, it is observed that the generalization success of the VGG-Face model is higher. Even though VGG-Face is a more straightforward model than TCAE and it does not include a unique mechanism for head pose variations, it captured better representations of the facial expressions by having more features. TCAE model provides 256 features, while VGG-Face provides 2622 features, which seemed to affect performances significantly. On the other hand, TCAE also proves its effectiveness with its disentanglement mechanism as a self-supervised learning model.

As future work, several improvements are considered. First, the reliability of the automatic coding must be proved with more samples by estimating the agreement between manual coders and trained models. To this end, more non-posed data samples from real-world conditions are needed to be collected. On the other hand, these samples can be included in the model training to improve the models' performances. Another critical thing to consider while collecting more samples is to balance the classes. Using weighted sampling mechanisms can also be included in the pipeline. Finally, the models' AU detection capability can be extended with some AUs such as AU2, AU7, AU20, AU25, and AU26 to grasp the children's expressions better.

To enhance the model performance, there is lots of room for improvement. First, the preprocessing methods can be upgraded using robust methods such as 3d registration, or patch learning. Also, recent methods such as patch attention networks or hybrid networks can be adapted to our system.

## REFERENCES


[1] **Benitez-Quiroz, C.F.**, **Srinivasan, R. and Martínez, A.** (2016). EmotioNet: An Accurate, Real-Time Algorithm for the Automatic Annotation of a Million Facial Expressions in the Wild, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5562–5570.

[2] **Oster, H.** (2000). Baby facs: Facial action coding system for infants and young children, *Unpublished monograph and coding manual*.

[3] **Baglayici, E.**, **Gurpinar, C.**, **Uluer, P. and Kose, H.** (2021). A New Facial Expression Processing System for an Affectively Aware Robot, ***A. Del Bimbo**, **R. Cucchiara**, **S. Sclaroff**, **G.M. Farinella**, **T. Mei**, **M. Bertini**, **H.J. Escalante and R. Vezzani**, editors, Pattern Recognition. ICPR International Workshops and Challenges*, Springer International Publishing, Cham, pp.36–51.

[4] **De la Torre, F.**, **Chu, W.S.**, **Xiong, X.**, **Vicente, F.**, **Ding, X. and Cohn, J.** (2015). IntraFace.

[5] **Standford**, CS231n, `https://cs231n.github.io/neural-networks-1/`, reference date: 19/06/2021.

[6] **Jadon, S.**, Introduction to Different Activation Functions for Deep Learning, `https://medium.com/@shrutijadon10104776`, reference date: 19/06/2021.

[7] **Marcelino, P.**, Transfer learning from pre-trained models, `https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751`, reference date: 19/06/2021.

[8] **Standley, T.S.**, **Zamir, A.**, **Chen, D.**, **Guibas, L.**, **Malik, J. and Savarese, S.** (2020). Which Tasks Should Be Learned Together in Multi-task Learning?, *ICML*.

[9] **Kovenko, V.**, How to precisely align face in Python with OpenCv and Dlib, `https://medium.com/@dsfellow/precise-face-alignment-with-opencv-dlib-e6c8acead262`, reference date: 19/06/2021.

[10] **Li, Y.**, **Zeng, J.**, **Shan, S. and Chen, X.** (2019). Self-Supervised Representation Learning From Videos for Facial Action Unit Detection, *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.10916–10925.

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

[12] **Friesen, W.V. and Ekman, P.** (1978). Facial action coding system: a technique for the measurement of facial movement, *Palo Alto*, *3*.

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

[14] **Chen, L.**, **Zhou, M.**, **Su, W.**, **Wu, M.**, **She, J. and Hirota, K.** (2018). Softmax regression based deep sparse autoencoder network for facial emotion recognition in human-robot interaction, *Information Sciences*, *428*, 49–61, `https://www.sciencedirect.com/science/article/pii/S0020025517310496`.

[15] **Hung, J.C.**, **Lin, K.C. and Lai, N.X.** (2019). Recognizing learning emotion based on convolutional neural networks and transfer learning, *Applied Soft Computing*, *84*, 105724, `https://www.sciencedirect.com/science/article/pii/S1568494619305058`.

[16] **Mayya, V.**, **Pai, R.M. and Manohara Pai, M.** (2016). Automatic Facial Expression Recognition Using DCNN, *Procedia Computer Science*, *93*, 453–461, `https://www.sciencedirect.com/science/article/pii/S1877050916314752`, proceedings of the 6th International Conference on Advances in Computing and Communications.

[17] **Guillén, S.S.**, **Iacono, L.L. and Meder, C.** (2018). Affective Robots: Evaluation of Automatic Emotion Recognition Approaches on a Humanoid Robot towards Emotionally Intelligent Machines, *World Academy of Science, Engineering and Technology, International Journal of Mechanical and Mechatronics Engineering*, *12*, 584–592.

[18] **Greco, A.**, **Roberto, A.**, **Saggese, A.**, **Vento, M. and Vigilante, V.** (2019). Emotion analysis from faces for social robotics, *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp.358–364.

[19] **Mollahosseini, A.**, **Chan, D. and Mahoor, M.H.** (2016). Going deeper in facial expression recognition using deep neural networks, *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp.1–10.

[20] **Rao, A.**, **Ajri, S.**, **Guragol, A.**, **Suresh, R. and Tripathi, S.** (2020). Emotion Recognition from Facial Expressions in Children and Adults Using Deep Neural Network, ***S.M. Thampi***, ***L. Trajkovic***, ***S. Mitra***, ***P. Nagabhushan***, ***E.S.M. El-Alfy***, ***Z. Bojkovic and D. Mishra***, *editors, Intelligent Systems, Technologies and Applications*, Springer Singapore, Singapore, pp.43–51.

[21] **Rudovic, O.**, **Utsumi, Y.**, **Lee, J.**, **Hernandez, J.**, **Ferrer, E.C.**, **Schuller, B. and Picard, R.W.** (2018). CultureNet: A Deep Learning Approach for Engagement Intensity Estimation from Face Images of Children with Autism, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp.339–346.

[22] **Zhao, K.**, **Chu, W.S. and Zhang, H.** (2016). Deep Region and Multi-label Learning for Facial Action Unit Detection, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3391–3399.

[23] **Ebrahimi Kahou, S.**, **Michalski, V.**, **Konda, K.**, **Memisevic, R. and Pal, C.** (2015). Recurrent Neural Networks for Emotion Recognition in Video, *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ICMI '15, Association for Computing Machinery, New York, NY, USA, p.467–474, `https://doi.org/10.1145/2818346.2830596`.

[24] **Breuer, R. and Kimmel, R.** (2017). A Deep Learning Perspective on the Origin of Facial Expressions, *ArXiv*, *abs/1705.01842*.

[25] **Ghosh, S.**, **Laksana, E.**, **Scherer, S. and Morency, L.P.** (2015). A multi-label convolutional neural network approach to cross-domain action unit detection, *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, pp.609–615.

[26] **Chu, W.S.**, **De la Torre, F. and Cohn, J.F.** (2017). Learning Spatial and Temporal Cues for Multi-Label Facial Action Unit Detection, *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pp.25–32.

[27] **Onal Ertugrul, I.**, **Yang, L.**, **Jeni, L.A. and Cohn, J.F.** (2019). D-PAttNet: Dynamic Patch-Attentive Deep Network for Action Unit Detection, *Frontiers in Computer Science*, *1*, 11, `https://www.frontiersin.org/article/10.3389/fcomp.2019.00011`.

[28] **Lim, Y.**, **Liao, Z.**, **Petridis, S. and Pantic, M.** (2018). Transfer Learning for Action Unit Recognition.

[29] **Tang, C.**, **Zheng, W.**, **Yan, J.**, **Li, Q.**, **Li, Y.**, **Zhang, T. and Cui, Z.** (2017). View-Independent Facial Action Unit Detection, *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pp.878–882.

[30] **Hammal, Z.**, **Chu, W.S.**, **Cohn, J.F.**, **Heike, C. and Speltz, M.L.** (2017). Automatic action unit detection in infants using convolutional neural network, *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, pp.216–221.

[31] **Li, W.**, **Abtahi, F.**, **Zhu, Z. and Yin, L.** (2017). EAC-Net: A Region-Based Deep Enhancing and Cropping Approach for Facial Action Unit Detection, *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pp.103–110.

[32] **Shao, Z.**, **Liu, Z.**, **Cai, J. and Ma, L.** (2018). Deep Adaptive Attention for Joint Facial Action Unit Detection and Face Alignment.

[33] **Wiles, O.**, **Koepke, A.S. and Zisserman, A.** (2018). Self-supervised learning of a facial attribute embedding from video, *BMVC*.

[34] **Albiero, V.**, **Bellon, O.R.P. and Silva, L.** (2018). Multi-Label Action Unit Detection on Multiple Head Poses with Dynamic Region Learning, *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp.2037–2041.

[35] **Jiang, Y.**, **Li, W.**, **Hossain, M.S.**, **Chen, M.**, **Alelaiwi, A. and Al-Hammadi, M.** (2020). A snapshot research and implementation of multimodal information fusion for data-driven emotion recognition, *Information Fusion*, *53*, 209–221.

[36] **Lucey, P.**, **Cohn, J.F.**, **Kanade, T.**, **Saragih, J.**, **Ambadar, Z. and Matthews, I.** (2010). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression, *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, IEEE, IEEE, San Francisco, CA, USA, pp.94–101.

[37] **He, K.**, **Zhang, X.**, **Ren, S. and Sun, J.** (2016). Deep Residual Learning for Image Recognition, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Las Vegas, NV, USA, pp.770–778.

[38] **Mavadati, S.M.**, **Mahoor, M.H.**, **Bartlett, K.**, **Trinh, P. and Cohn, J.F.** (2013). DISFA: A Spontaneous Facial Action Intensity Database, *IEEE Transactions on Affective Computing*, *4*(2), 151–160.

[39] **LoBue, V. and Thrasher, C.** (2015). The Child Affective Facial Expression (CAFE) set: Validity and reliability from untrained adults, *Frontiers in psychology*, *5*, 1532.

[40] **Li, L.**, **Yuan, Y.**, **Li, M.**, **Xu, H.**, **Li, R. and Lu, S.** (2019). Subject Independent Facial Expression Recognition: Cross-Connection and Spatial Pyramid Pooling Convolutional Neural Network, *IVSP 2019*.

[41] **Zhang, X.**, **Yin, L.**, **Cohn, J.**, **Canavan, S.**, **Reale, M.**, **Horowitz, A.**, **Liu, P. and Girard, J.** (2014). BP4D-Spontaneous: A high-resolution spontaneous 3D dynamic facial expression database, *Image and Vision Computing*, *32*, 692–706.

[42] **Fan, R.E.**, **Chang, K.W.**, **Hsieh, C.J.**, **Wang, X.R. and Lin, C.J.** (2008). LIBLINEAR: a library for large linear classification, *Journal of Machine Learning Research*, *9*, 1871–1874.

[43] **Jain, N.**, **Kumar, S.**, **Kumar, A.**, **Shamsolmoali, P. and Zareapoor, M.** (2018). Hybrid deep neural networks for face emotion recognition, *Pattern Recognition Letters*, *115*, 101–106, `https://www.sciencedirect.com/science/article/pii/S0167865518301302`, multimodal Fusion for Pattern Recognition.

[44] **Parkhi, O.M.**, **Vedaldi, A. and Zisserman, A.** (2015). Deep Face Recognition, ***X. Xie**, **M.W. Jones and G.K.L. Tam**, editors, Proceedings of the British Machine Vision Conference (BMVC)*, BMVA Press, Swansea, UK, pp.41.1–41.12, `https://dx.doi.org/10.5244/C.29.41`.

[45] **Nagrani, A.**, **Chung, J.S. and Zisserman, A.** (2017). VoxCeleb: a large-scale speaker identification dataset, *INTERSPEECH*.

[46] **Baltrusaitis, T.**, **Zadeh, A.**, **Lim, Y.C. and Morency, L.P.** (2018). Openface 2.0: Facial behavior analysis toolkit, *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, IEEE, IEEE, Xi'an, China, pp.59–66.

[47] **Neyshabur, B.**, **Sedghi, H. and Zhang, C.** (2020). What is being transferred in transfer learning?, ***H. Larochelle**, **M. Ranzato**, **R. Hadsell**, **M.F. Balcan and H. Lin**, editors, Advances in Neural Information Processing Systems*, volume 33, Curran Associates, Inc., pp.512–523, https://proceedings.neurips.cc/paper/2020/file/0607f4c705595b911a4f3e7a127b44e0-Paper.pdf.

[48] **Yosinski, J.**, **Clune, J.**, **Bengio, Y. and Lipson, H.** (2014). How transferable are features in deep neural networks?, ***Z. Ghahramani**, **M. Welling**, **C. Cortes**, **N. Lawrence and K.Q. Weinberger**, editors, Advances in Neural Information Processing Systems*, volume 27, Curran Associates, Inc., https://proceedings.neurips.cc/paper/2014/file/375c71349b295fbe2dcdca9206f20a06-Paper.pdf.

[49] **H. Köse**, **D.E. Barkana**, **H.E.S.Y.**, "An Affectively Aware Assistive Robotic Platform for Rehabilitation of Deaf Children", Symposium on Robots for Language Learning, Istanbul, Turkey, December 2018, https://roborehab.itu.edu.tr/index.html, reference date: 19/06/2021.

[50] **Dalrymple, K.A.**, **Gomez, J. and Duchaine, B.** (2013). The Dartmouth Database of Children's Faces: Acquisition and Validation of a New Face Stimulus Set, *PLoS ONE*, *8*.

[51] **Khan, R.A.**, **Crenn, A.**, **Meyer, A. and Bouakaz, S.** (2019). A novel database of children's spontaneous facial expressions (LIRIS-CSE), *Image and Vision Computing*, *83-84*, 61–69, https://www.sciencedirect.com/science/article/pii/S0262885619300137.

[52] **Li, S. and Deng, W.** (2020). Deep Facial Expression Recognition: A Survey, *IEEE Transactions on Affective Computing*, 1–1.

[53] **opencv**, OpenCV - Face detection module, https://github.com/opencv/opencv/tree/master/samples/dnn, reference date: 19/06/2021.

[54] **King, D.E.** (2009). Dlib-ml: A Machine Learning Toolkit, *J. Mach. Learn. Res.*, *10*, 1755–1758.

[55] **Zhu, X.**, **Liu, Y.**, **Li, J.**, **Wan, T. and Qin, Z.** (2018). Emotion Classification with Data Augmentation Using Generative Adversarial Networks, *PAKDD*.

**CURRICULUM VITAE**

**Name Surname** : **Süleyman Engin Bağlayıcı**

**EDUCATION** :

- **B.Sc.** : 2017, Istanbul Gedik University, Engineering Faculty, Mechatronics Engineering

**PROFESSIONAL EXPERIENCE AND REWARDS:**

- 2018-Present Software Engineer at AVL
- 2017-2018 Robotics Systems R&D Engineer at Gedik Robotics
- 2017 2nd Place in Remotely Operated Underwater Vehicle Competition of MATE ROV Turkey

**PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:**

- **Baglayici E., Gurpinar C., Uluer P., Köse H.** 2021. A New Facial Expression Processing System for an Affectively Aware Robot. *Computational & Affetive Intelligence in Healthcare Applications for Vulnerable Populations Workshop*, January 10, 2021, Online Streaming.

**OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:**

- **Çaldıran O., Bağlayıcı E., Dousti M., Mungan E., Bulut E.E., Demir M.D., Koçyiğit F.** 2021. Automated Lane Change Decision Making in Highway Using a Hybrid Approach. *International Conference on Informatics in Control, Automation and Robotics*, July 7, 2021, Online Streaming.

- **Akkaya Ş., Gürbüz Y., Zile M.G., Bağlayıcı E., Şeker H.A., Erdoğan A.** 2019. A Modular Five-Layered V-Shaped Architecture for Autonomous Vehicles. *11th International Conference on Electrical and Electronics Engineering*, November 28, 2019 Bursa, Turkey.