BABEŞ BOLYAI UNIVERSITY, CLUJ NAPOCA, ROMÂNIA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Face expression recognition for social good
- road to smart acting -

– ITSG report –

Teacher
Laura Dioșan
Intelligent Tools for Social Good

Author
Bogdan-Daniel Bălănescu
Software Engineering, 258-1

2019

Abstract

This paper studies the problem of Face Expression Recognition (REC) in an attempt to build a tool that helps novice actors better analyze their performance and get real-time feedback to improve. Starting from the Convolutional Neural Network proposed in [1], and trained on the FER-2013 emotion database, this paper slightly improves the 66% learning accuracy of the proposed algorithm to 68%. An application, Acting Mirror, is proposed and presented in the last part of the paper, as well as all the results and comparisons with other state of the art research. In the conclusion, we see that the current study of FER is facing a few issues we raise from our study and we propose a few improvement action points that could be tackled in the future.

Contents

1 Introduction **1**
- 1.1 What? Why? How? . . . 1
- 1.2 Paper structure and original contribution(s) . . . 2

2 Scientific Problem **3**
- 2.1 Problem definition . . . 3

3 State of art/Related work **5**
- 3.1 State of the Art . . . 5
- 3.2 Useful Tools . . . 6

4 Proposed approach **8**
- 4.1 Algorithm description . . . 8

5 Application (numerical validation) **10**
- 5.1 Methodology . . . 10
- 5.2 Obtained results . . . 11
- 5.3 Case for improvement . . . 26

6 Acting Mirror **40**

7 Conclusion and future work **42**
- 7.1 State of the Art Comparison . . . 42
- 7.2 Brief Ethical Discussion . . . 43
- 7.3 Acting Mirror and Future Work . . . 45

Chapter 1

Introduction

1.1 What? Why? How?

For novice actors, it is often hard to control their emotions and to express what is on their script well, so the need for an expert to be with them when they rehearse is needed. A tool that would give them feedback in real-time may just make their start easier. It could be there for them anytime and for free, rather than hiring an expert to watch them rehearse.
This paper addresses this problem by proposing a tool able to recognize emotions from real-time video footage or photos.

- ***What?*** The problem of Face Expression Recognition (FER) is a classification problem which has received an important amount of attention in the last decade with various approaches such as the feature-based Tree-Augmented-Naive Bayes (TAN) classifier, Local Binary Patterns (LBP) classifier, Support Vector Machines and numerous Neural Networks based approaches. [3]

- ***Why?*** The importance of FER is present in a variety of domains, such as: psychology, neuroscience and even philosophy. Science today still cannot explain for sure where do emotions come from or if emotions are the ones driving our decisions or not. With this in mind, an approach that involves artificial intelligence capable of classifying emotions from photos would prove useful until other studies provide a better or a more accessible solution.

- ***How?*** In this paper, we present an artificially intelligent solution to recognizing emotions from pictures. From a dataset of pictures labeled with emotions we will train a model able to classify emotions.

[The work of next laboratories will add here: A short discussion of how it fits into related work in

the area. Summary of the basic results and conclusions.]

1.2 Paper structure and original contribution(s)

[Should present here this information when improvement of the algorithm is done: The research presented in this paper advances the theory, design, and implementation of the proposed alogrithm.]

The main contribution of this report is to present an intelligent algorithm for solving the problem of Face Expression Recognition.

The second contribution of this report consists of building an intuitive, easy-to-use and user friendly software application. Our aim is to build an application that will help novice actors to better analyze their facial expressions and get real-time feedback while rehearsing. Main feature present in the application (and other possible improvements):

- ***Main feature:*** The user shall be able to record themselves in real-time and see their emotions in the video.
- ***Future improvement:*** The user shall be able to provide an acting script with labeled emotions on certain sections of text.
- ***Future improvement:*** The user shall receive real-time feedback based on the labeled script when they make a mistake.

The third contribution of this thesis consists of providing a comparison between state of the art results in FER and the proposed algorithm.

[Should be present in the next laboratories: The present work contains xyz bibliographical references and is structured in five chapters as follows.]

In the second chapter we will take a look at a formal introduction of our FER problem and weight the advantages and disadvantages of using AI to solve it.

The third chapter will describe the state of the art in FER.

In the the fourth chapter we will further detail our proposed approach to solving the problem.

[Short placeholder here, until we reach detailing the third chapter: (dataset with pictures and labels (emotion) -> new dataset facial points (features) and labels (emotion) -> model to solve the problem)]

The fifth chapter will present a comparison for the chosen methodology, data and results between two different approaches to solving this problem (ours and one more).

The final chapter will present our conclusions and future work to be done regarding this problem and application.

Chapter 2

Scientific Problem

++

2.1 Problem definition

For people pursuing their hobby of acting, it might be hard to hire an expert while rehearsing or even coming to terms with the tight schedule of a hard working day and a few minutes of spare time to rehearse a play. The need for a tool that would assist people when rehearsing, giving them feedback about their facial expressions in real life, arises and we pursue to deliver such an application.
The solution to such a tool is required to use an intelligent algorithm, because as far as we know, there exist no other methodologies for approaching this problem and it also falls into the category of complex problems that may be more easily solved using a neural network or other intelligent algorithms.

Advantages of solving the problem with an intelligent algorithm:

- it is much faster to reach a solution than proceeding with finding a non-intelligent algorithm to solve it
- it may be impossible to solve it using non-intelligent algorithms due to the vast lack of knowledge in the area of human emotions

Disadvantages of solving the problem with an intelligent algorithm:

- it may require a lot of work in training and retraining models until we reach a suitable (efficient and accurate) model to solve the problem
- it is impossible to reach, using A.I., a solution which has a 100% accuracy

Short description of our initial approach:

- From an existing dataset, FER-2013, which contains 35,887 48x48 pixel grayscale labeled images, 80% of pictures were used for training, while the remaining 20% were used for validation of the algorithm.
- The model was trained locally using Keras on an i7-7500u and an NVIDIA Quadro M520 1GB. It took a little bit over 2 hours of training and the achieved validation accuracy was 65%. While doing the validation precision/accuracy test, it takes approximately 1 milliseconds for the model to analyze a 48x48 pixel grayscale image.

Chapter 3

State of art/Related work

3.1 State of the Art

In this section we present a few methods utilized in order to solve the Facial Expression Recognition (FER) problem.

First, we will take a look at [7], where the problem is solved using k-NN (Nearest Neighbors) and MLP (Multilayer Perceptron).

- ***What kind of data did they use?*** Coefficients describing elements of facial expressions (as features) and a range of seven emotional states (as labels). The emotional states detected are: neutral, joy, sadness, surprise, anger, fear, disgust. The images they used were from the KDEF database.

- ***How does their model(s) work?*** Using Microsoft Kinect 3D for face modeling, they were able to extract 3D models of the face as 3D points, but Kinect 3D also can extract Action Units (AC) based on those points, which basically represent certain features of the face. Choosing 6 of those Action Units (upper lip raising, jaw lowering, lip stretching, lowering eyebrows, lip corner depressing, outer brow raising) they were able to train a 3-NN and an MLP classifier in order to solve the FER problem.

- ***What were their results?*** They tested the models for two cases: a) subject-dependent and b) subject-independent. For the 3-NN classifier, they got around 95-96% accuracy and for the MLP algorithms the results were around 75-76%.

Second, let us look at [6], where three significant challenges in FER are discussed: illumination variation, head pose and subject-dependence. We shall focus on the ones that target visual-only databases.

- ***What about illumination variation?*** The paper compares different approaches to FER, like SVM (Support Vector Machines) with 31-50% accuracy, Deep Networks with 48-96% accuracy and KNN with 92-96% accuracy. The most utilized dataset was the CK+ dataset. The paper suggests using Fast Fourier Transform and Contrast Limited Adaptive Histogram Equalization (FFT+CLAHE) to overcome poor lighting conditions, among other techniques.

- ***What about subject-dependece?*** Subject-dependence means the model is only able to recognize the expressions of the faces it trained with. The paper proposes a solution, where geometric face features were extracted, and using a part-based hierarchical recurrent neural network (PHRNN) to model the facial morphological variations, a multi-signal convolutional neural network (MSCNN) to find the spatial features of face, an accuracy of around 98.5% can be achieved. The used dataset was CK+.

Last, but not least, let us take a look at [2], a practical approach using a Convolutional Neural Network (CNN).

- ***What kind of data did they use?*** The used datasets were MMI and CKP and they recognized emotions from the following list: anger, sadness, disgust, happiness, fear and surprise.

- ***How does their model(s) work?*** Their proposed model is independent from any other third-party feature extraction frameworks and it performs better than the previously proposed CNN models, with an accuracy of 93-99% on the above mentioned datasets. Their model begins with a convolutional layer over the input and then filtering max pooling layer, before entering two fully connected layers (comprised of convolution + pooling, then convolution + convolution, then concatenation and pooling), after which they classify the images with softmax layer.

- ***What were their results?*** Their accuracy was around 99%.

So far, we have seen three great state of the art examples. One which uses an third-party framework to manipulate the dataset so that the model can be a very simple one to train, like a 3-NN. Another set of examples where top state of the art models were presented and certain impediments were discussed (such as illumination variation and subject-dependency) and one other example which is independent on any other third-party framework in its learning, while still achieving good results.

3.2 Useful Tools

Now, let us give a list of useful tools to use when developing intelligent applications:

- Tensorflow, is a Python framework for building machine learning models. A javascript version, Tensorflow.js, also exists.
- Tensorflow Lite, a framework for building machine learning models compatible with portable devices (i.e. mobile phones).
- ML Kit for Firebase, a mobile SDK that empowers mobile applications with Google's machine learning packages. It is also possible to host one's own machine learning model in Firebase (have not experienced with this yet).
- YOLO: Real-Time Object Detection, is a state of the art object detection system, but which can also be trained for a different purpose. It is open source and written in C++.
- fastAI, is a Python framework for buildign machine learning models. Also comes with pre-trained models for certain problems.

Chapter 4

Proposed approach

4.1 Algorithm description

Starting from the Convolutional Neural Network proposed in [1] (and which is inspidred by the Xception [4] architecture), and presented in the following figure (see Figure 4.1), our approach is to use an already existing algorithm capable of classifying emotions and integrate it in an easy to use application in order to help novice actors get real-time feedback about their performance.

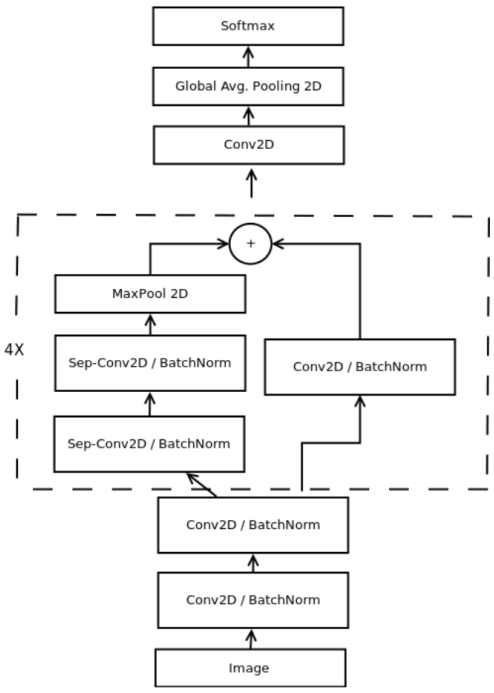

Figure 4.1: [1] proposed model for real-time classification of emotions

We used the FER-2013 emotion database, which contains 35.887 48x48 pixel grayscale labeled

images with the following emotions: (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). Initially, we performed 7 training trials, one using original approach proposed in [1], and then by changing the learning rate and the loss function during training.

For the original approach proposed in [1], we achieved a 65% validation accuracy over the FER-2013 database (using all the 35.887 images), and a 76% accuracy over the CK+ database using the same model trained on the FER-2013 database.

From our other approaches, the fifth one achieved a 64% validation accuracy over the FER-2013 dabase (using only 28.709 of the images), and a 79% accuracy over the CK+ database using the same model trained on the FER-2013 database.

Chapter 5

Application (numerical validation)

For the proposed algorithm, our approach was to perform several training sessions, with different inputs for the independent variables and see how they affect the learning. The reason for this, was to find a way to improve the accuracy of the model. As will be related in the following section, we have played around with the learning rate and with the loss function of the algorithm. Unfortunately, what we have found is that the algorithm reaches a plateou of learning from which it cannot escape, thus, the accuracy of the model being around 65%, almost like what was achieved in [1].

5.1 Methodology

In this chapter, we analyze the achieved results for the proposed algorithm using the following approach. We present the training dataset and independent variables, a plot of the achieved training results and how well it fairs against the CK+ dataset, presenting the precision for each emotion separately.
The independent variables we are experimenting with are the learning rate, and the loss function.
The dependent variables will be the weights of the model.
For our first attempt, we have uesd the whole FER-2013 dataset, and obtained a validation accuracy of 65%, and for the next attempts we shall only use the Training pictures from the FER-2013 dataset, and it will also enable us to compute the precision for the Public Test set nad Private Test set of the FER-2013 dataset.
For comparison purposes between our models trained on the FER-2013 dataset, we also compute the precision for each model against the CK+ emotion dataset and present the results here.

5.2 Obtained results

The first trial:

- **Dataset**: FER-2013, 35.887 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Categorical Crossentropy.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a training validation accuracy of 65%, and a 76% accuracy on the CK+ dataset.

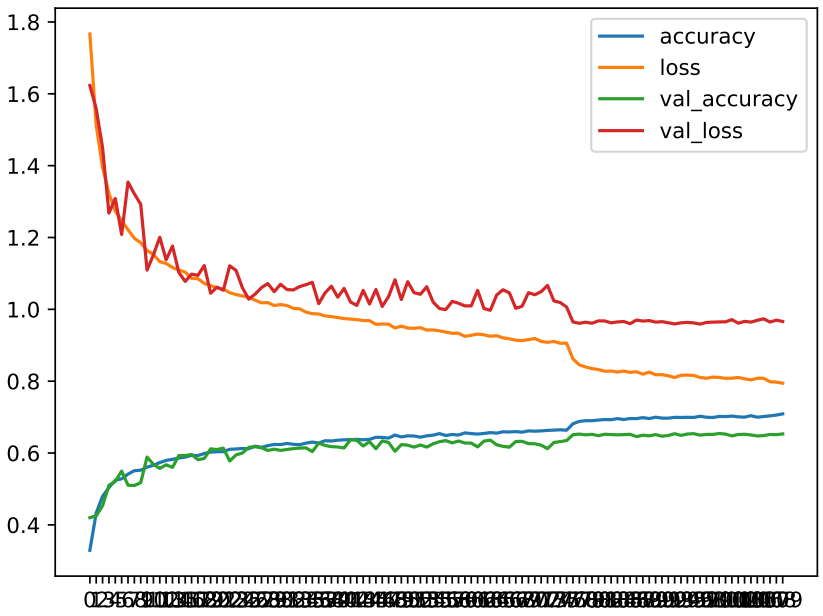

Figure 5.1: Training results using all the 35.887 images in the FER-2013 emotion dataset

Table 5.1: The precisin results against the CK+ dataset for the first training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.30	45
Disgust	0.91	59
Fear	0.43	25
Happy	0.94	69
Sad	0.39	28
Surprise	0.97	83
Neutral	0.00	1
Weighted Average	0.76	310

The second trial:

- **Dataset**: FER-2013, 28.709 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Categorical Crossentropy.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a training validation accuracy of 63%, and a 76% accuracy on the CK+ dataset.

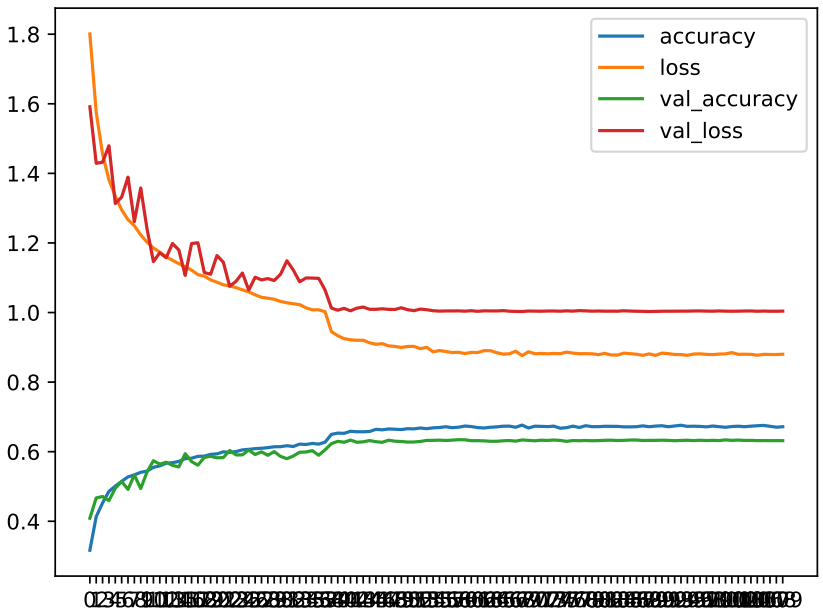

Figure 5.2: Training results using 28.709 images from the FER-2013 emotion dataset

Table 5.2: The precisin results against the CK+ dataset for the second training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.21	45
Disgust	1.00	59
Fear	0.53	25
Happy	0.91	69
Sad	0.52	28
Surprise	0.94	83
Neutral	0.00	1
Weighted Average	0.76	310

Table 5.3: The precisin results against the FER-2013 Public Test dataset for the second training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.51	467
Disgust	0.67	56
Fear	0.47	496
Happy	0.82	895
Sad	0.52	653
Surprise	0.74	415
Neutral	0.55	607
Weighted Average	0.62	3589

Table 5.4: The precisin results against the FER-2013 Private Test dataset for the second training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.53	467
Disgust	0.47	56
Fear	0.47	496
Happy	0.84	895
Sad	0.50	653
Surprise	0.74	415
Neutral	0.61	607
Weighted Average	0.63	3589

The third trial:

- **Dataset**: FER-2013, 28.709 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Categorical Crossentropy.
- **Optimizer**: Adam.
- **Learning rate**: 0.01.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a validation accuracy of 63%, and a 60% accuracy on the CK+ dataset.

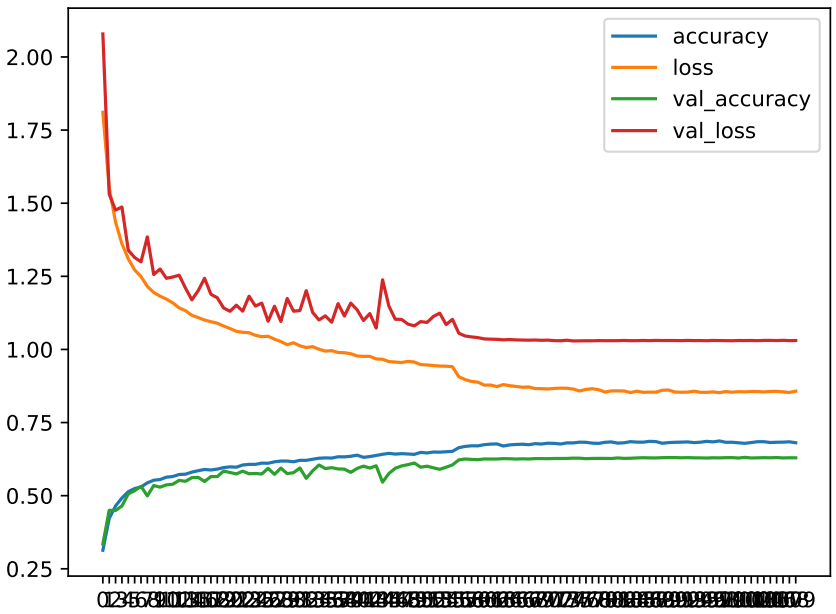

Figure 5.3: Training results using 28.709 images from the FER-2013 emotion dataset

Table 5.5: The precisin results against the CK+ dataset for the second training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.25	45
Disgust	0.00	59
Fear	0.80	25
Happy	0.89	69
Sad	0.54	28
Surprise	0.95	83
Neutral	0.00	1
Weighted Average	0.60	310

Table 5.6: The precisin results against the FER-2013 Public Test dataset for the third training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.52	467
Disgust	0.62	56
Fear	0.48	496
Happy	0.84	895
Sad	0.54	653
Surprise	0.74	415
Neutral	0.55	607
Weighted Average	0.63	3589

Table 5.7: The precisin results against the FER-2013 Private Test dataset for the third training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.57	467
Disgust	0.73	56
Fear	0.48	496
Happy	0.85	895
Sad	0.48	653
Surprise	0.76	415
Neutral	0.59	607
Weighted Average	0.64	3589

The fourth trial:

- **Dataset**: FER-2013, 28.709 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Mean Square Error.
- **Optimizer**: Adam.
- **Learning rate**: 0.01.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result were a validation accuracy of 63%, and a 69% accuracy on the CK+ dataset.

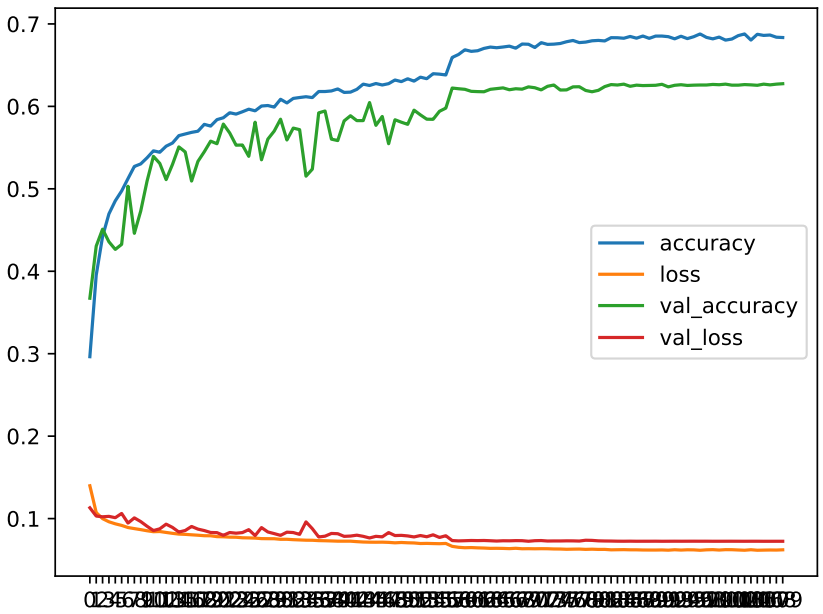

Figure 5.4: Training results using 28.709 images from the FER-2013 emotion dataset

Table 5.8: The precisin results against the CK+ dataset for the fourth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.23	45
Disgust	0.67	59
Fear	0.56	25
Happy	0.85	69
Sad	0.43	28
Surprise	0.96	83
Neutral	0.00	1
Weighted Average	0.69	310

Table 5.9: The precisin results against the FER-2013 Public Test dataset for the fourth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.53	467
Disgust	0.61	56
Fear	0.51	496
Happy	0.82	895
Sad	0.54	653
Surprise	0.76	415
Neutral	0.53	607
Weighted Average	0.63	3589

Table 5.10: The precisin results against the FER-2013 Private Test dataset for the fourth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.56	467
Disgust	0.52	56
Fear	0.50	496
Happy	0.85	895
Sad	0.49	653
Surprise	0.74	415
Neutral	0.58	607
Weighted Average	0.63	3589

The fifth trial:

- **Dataset**: FER-2013, 28.709 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Mean Square Error.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result were a validation accuracy of 64%, and a 79% accuracy on the CK+ dataset.

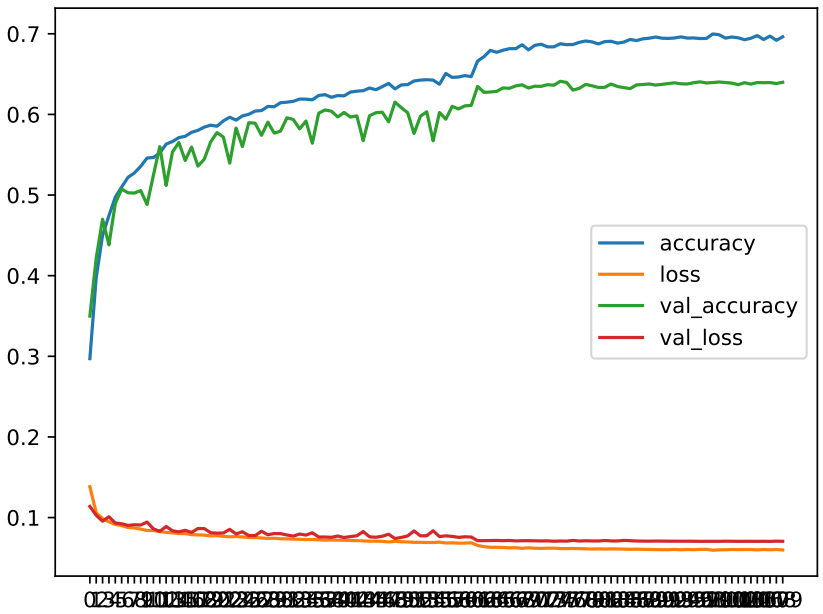

Figure 5.5: Training results using 28.709 images from the FER-2013 emotion dataset

Table 5.11: The precisin results against the CK+ dataset for the fifth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.21	45
Disgust	1.00	59
Fear	0.67	25
Happy	0.94	69
Sad	0.55	28
Surprise	0.94	83
Neutral	0.00	1
Weighted Average	0.79	310

Table 5.12: The precisin results against the FER-2013 Public Test dataset for the fifth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.53	467
Disgust	0.56	56
Fear	0.48	496
Happy	0.84	895
Sad	0.56	653
Surprise	0.78	415
Neutral	0.53	607
Weighted Average	0.63	3589

Table 5.13: The precisin results against the FER-2013 Private Test dataset for the fifth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.54	467
Disgust	0.62	56
Fear	0.47	496
Happy	0.87	895
Sad	0.49	653
Surprise	0.76	415
Neutral	0.58	607
Weighted Average	0.64	3589

The sixth trial:

- **Dataset**: FER-2013, 28.709 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Squared Hinge Loss.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result were a validation accuracy of 63%, and a 57% accuracy on the CK+ dataset.

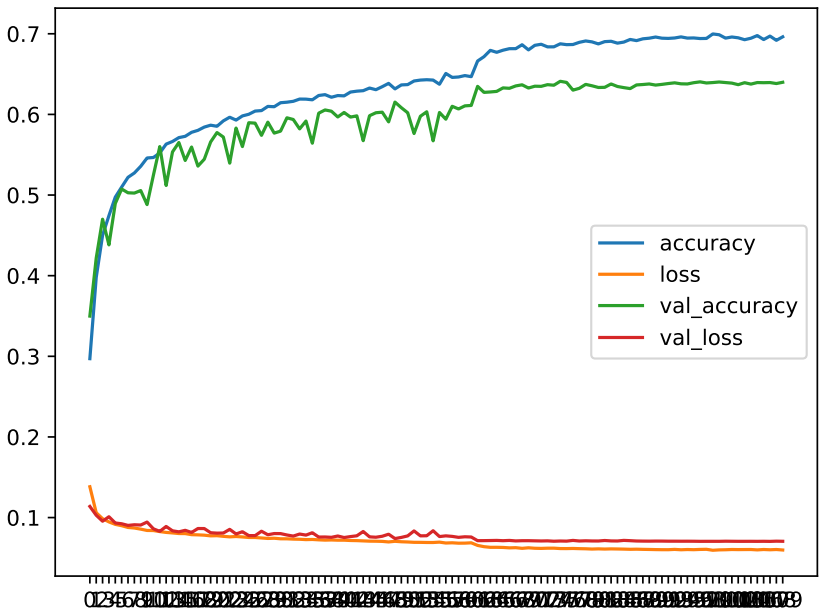

Figure 5.6: Training results using 28.709 images from the FER-2013 emotion dataset

Table 5.14: The precisin results against the CK+ dataset for the sixth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.23	45
Disgust	0.00	59
Fear	0.60	25
Happy	0.89	69
Sad	0.41	28
Surprise	0.95	83
Neutral	0.00	1
Weighted Average	0.57	

Table 5.15: The precisin results against the FER-2013 Public Test dataset for the sixth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.49	467
Disgust	0.00	56
Fear	0.51	496
Happy	0.85	895
Sad	0.54	653
Surprise	0.73	415
Neutral	0.53	607
Weighted Average	0.62	3589

Table 5.16: The precisin results against the FER-2013 Private Test dataset for the sixth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.52	467
Disgust	0.00	56
Fear	0.50	496
Happy	0.87	895
Sad	0.50	653
Surprise	0.72	415
Neutral	0.59	607
Weighted Average	0.63	3589

The seventh trial:

- **Dataset**: FER-2013, 28.709 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Squared Hinge Loss.
- **Optimizer**: Adam.
- **Learning rate**: 0.01.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result were a validation accuracy of 63%, and a 58% accuracy on the CK+ dataset.

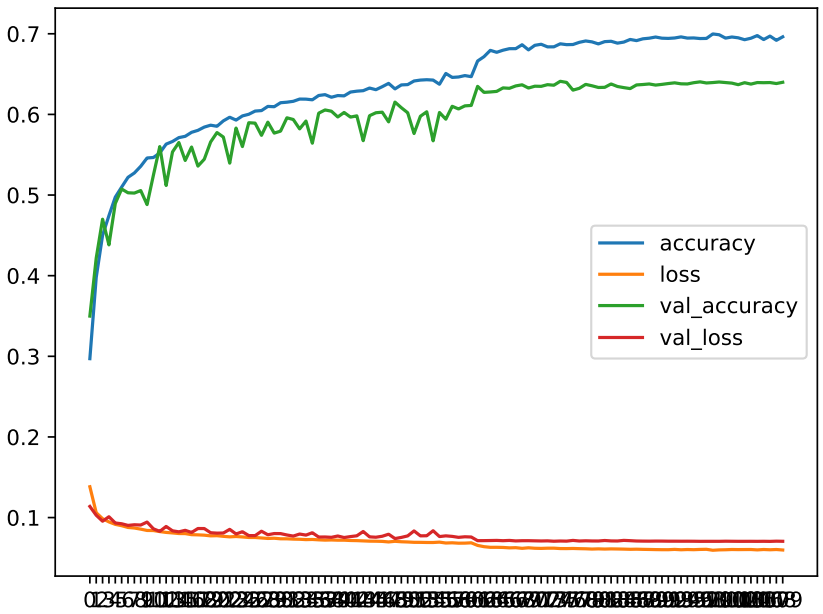

Figure 5.7: Training results using 28.709 images from the FER-2013 emotion dataset

Table 5.17: The precisin results against the CK+ dataset for the seventh training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.25	45
Disgust	0.00	59
Fear	0.50	25
Happy	0.92	69
Sad	0.46	28
Surprise	0.97	83
Neutral	0.00	1
Weighted Average	0.58	310

Table 5.18: The precisin results against the FER-2013 Public Test dataset for the seventh training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.50	467
Disgust	0.00	56
Fear	0.46	496
Happy	0.82	895
Sad	0.53	653
Surprise	0.71	415
Neutral	0.51	607
Weighted Average	0.60	3589

Table 5.19: The precisin results against the FER-2013 Private Test dataset for the seventh training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.53	467
Disgust	0.50	56
Fear	0.48	496
Happy	0.85	895
Sad	0.49	653
Surprise	0.71	415
Neutral	0.57	607
Weighted Average	0.62	3589

Analyzing the current situation:

While the achieved results were not a big improvements over the original approach, by changing the loss function from Categorical Crossentropy to Square Mean Error, we have managed to raise the accuracy of the model against the CK+ dataset from 76% (accuracy that is obtained by training either on just 28.709 of the images or on all 35.887 images from the FER-2013 dataset) to 79% (trained on just only 28.709 of the images from the FER-2013 datast). In the following attempts we aim to augment the dataset, so that it contains even more images and possibly achieve better results.

After these 7 trials we have noticed that all the models encounter problems at identifying the Angry emotion over the CK+ database. A point for improvement could be to find out whether the Angry emotion from the FER-2013 dataset gets confused with another emotion. We could do this either by training the model for just the Angry emotion and another one (Fear, Sad), or by eliminating the Angry emotion from the dataset and see if the precision for emotions Fear and Sad improves.

Another identified problem is that on certain trials, the emotion Disgust is not recognized at all within the CK+ dataset. Could it be that Disgust gets confused with another emotion, like Angry? Further investigation is needed.

Another observation could be that training with a high learning rate from the beginning may cause the model to reach a plateau much sooner. Lowering the learning rate seems to delay that.

Also, chaning the loss function from Categorial Crossentropy to Mean Square Error yielded improvements in the model later on (as in later epochs) as well, instead of stopping to learn since epoch 70-90.

Our best attempt was the fifth one: which uses Square Mean Error as the loss function and a learning rate of 0.1. Subsequent trials will use this setup.

5.3 Case for improvement

The eight trial - augmented dataset x6:

- **Dataset**: Augmented FER-2013 x6, 172.254 images, split into 80% training and 20% validation sets. Each of the below points describes an operation made for 28.701 images from the FER-2013 dataset using a python library called imagaug, [5]:
  - **Original**: 28.709 images from the original FER-2013 dataset.
  - **Additive Gaussian Noise**: with a scale of 0.07*255.
  - **Multiply**: with random values between 0.25 and 1.50.
  - **Salt and Pepper**: with a percentage value of 0.03.
  - **Gaussian Blur**: with a value of 0.50.
  - **Clouds**: for each image a cloud like mask was put over.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 7] tensor containing the percentages for each label.
- **Loss function**: Mean Square Error.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a training validation accuracy of 68%, and a 71% accuracy on the CK+ dataset.

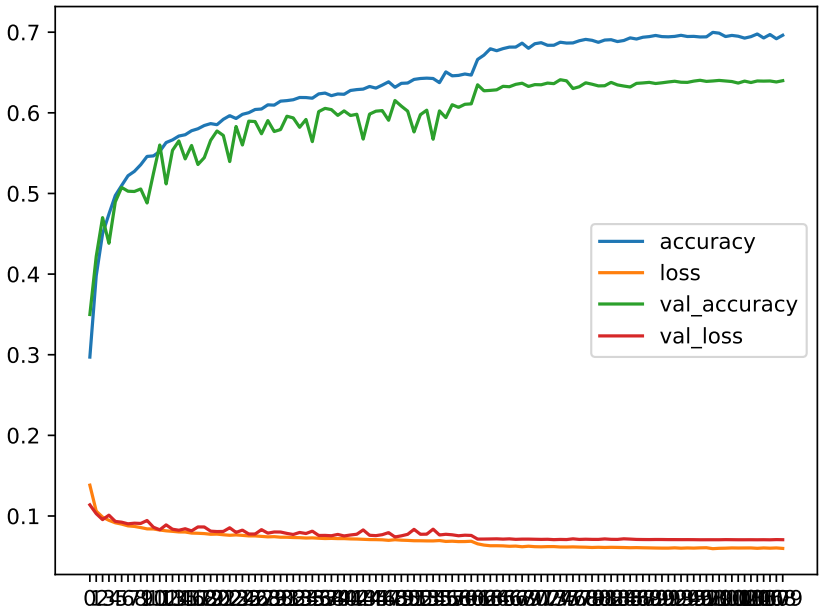

Figure 5.8: Training results using 172.254 augmented images from the FER-2013 emotion dataset

Table 5.20: The precisin results against the CK+ dataset for the eight training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.18	45
Disgust	0.86	59
Fear	0.33	25
Happy	0.89	69
Sad	0.38	28
Surprise	0.96	83
Neutral	0.00	1
Weighted Average	0.71	310

Table 5.21: The precisin results against the FER-2013 Public Test dataset for the eight training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.51	467
Disgust	0.62	56
Fear	0.51	496
Happy	0.83	895
Sad	0.56	653
Surprise	0.74	415
Neutral	0.57	607
Weighted Average	0.64	3589

Table 5.22: The precisin results against the FER-2013 Private Test dataset for the eight training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.55	467
Disgust	0.57	56
Fear	0.52	496
Happy	0.85	895
Sad	0.50	653
Surprise	0.75	415
Neutral	0.63	607
Weighted Average	0.65	3589

Analyzing the current situation:

There does not seem to be much improvement, apart from the increase on the validation accuracy against FER-2013 from 64% to 68%, compared to our fifth and most successful trial. But this improvement comes with a cost on the validation accuracy against the CK+ dataset from 79% to 71%. From the observation that on the CK+ dataset emotions Angry, Fear and Sad have an unreliable precision, we decide to build a Confusion Matrix for the dataset we trained on, FER-2013.

Table 5.23: The confusion matrix for FER-2013 Public Test dataset for the eigth training trial

Emotions	**Angry**	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Angry	**0.59**	0.01	0.09	0.06	0.10	0.03	0.10
Disgust	0.27	**0.41**	0.09	0.05	0.11	0.18	0.05
Fear	0.12	0.01	**0.43**	0.04	0.21	0.09	0.11
Happy	0.03	0.01	0.02	**0.85**	0.02	0.02	0.05
Sad	0.15	0.01	0.09	0.05	**0.51**	0.02	0.17
Surprise	0.03	0.01	0.08	0.06	0.02	**0.78**	0.03
Neutral	0.08	0.01	0.07	0.09	0.14	0.02	**0.61**

Table 5.24: The confusion matrix for FER-2013 Private Test dataset for the eigth training trial

Emotions	**Angry**	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Angry	**0.61**	0.02	0.09	0.03	0.15	0.01	0.09
Disgust	0.31	**0.45**	0.07	0.02	0.13	0.02	0.00
Fear	0.15	0.01	**0.43**	0.04	0.19	0.10	0.09
Happy	0.03	0.01	0.02	**0.88**	0.02	0.02	0.03
Sad	0.11	0.01	0.11	0.07	**0.52**	0.01	0.18
Surprise	0.05	0.00	0.10	0.04	0.03	**0.76**	0.02
Neutral	0.06	0.01	0.05	0.08	0.16	0.03	**0.63**

From the confusion matrixes we observe that there is high confusion between Agnry and Disgust. We decide to investigate further, and what we find is that the distribution of data across the 7 emotions proposed in FER-2013 is uniform. In 5.9 we observe that Disgust makes only 2% of the FER-2013 dataset and that the other emotions each make an average of 16-17% of the dataset. For this reason, we believe that excluding the Disgust emotion from our training dataset may improve the overall accuracy of the model. The following will be 2 trials training the model on the original, and respectively on the augmented version, of the FER-2013 dataset. The attempts will use the setup discussed in the fifth trial.

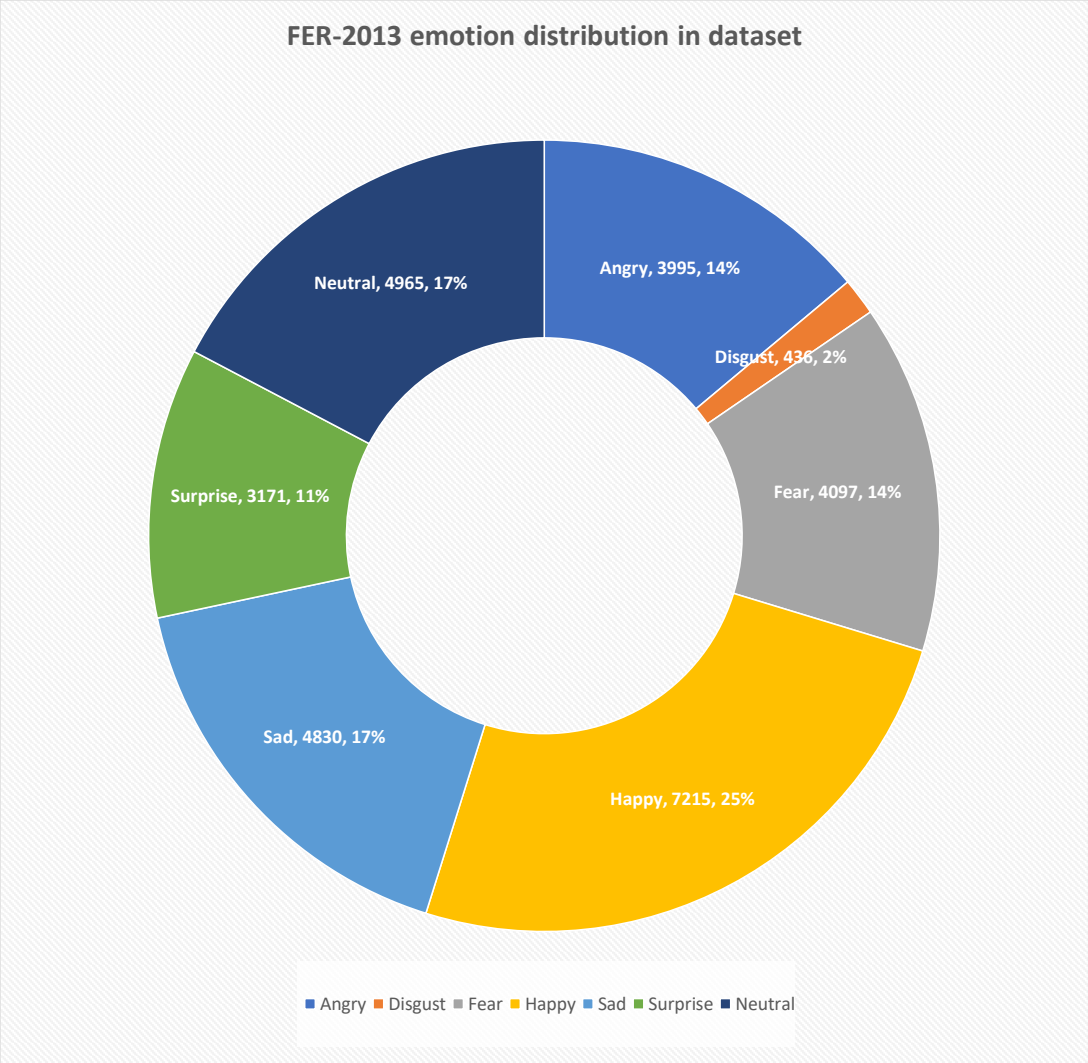

Figure 5.9: FER-2013 emotion distribution in dataset for the training images

The ninth trial - original dataset without the emotion "Disgust":

- **Dataset**: Original FER-2013 dataset, with the emotion "Disgust" removed, 28.273 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 6] tensor containing the percentages for each label.
- **Loss function**: Mean Square Error.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a training validation accuracy of 65%, and a 81% accuracy on the CK+ dataset.

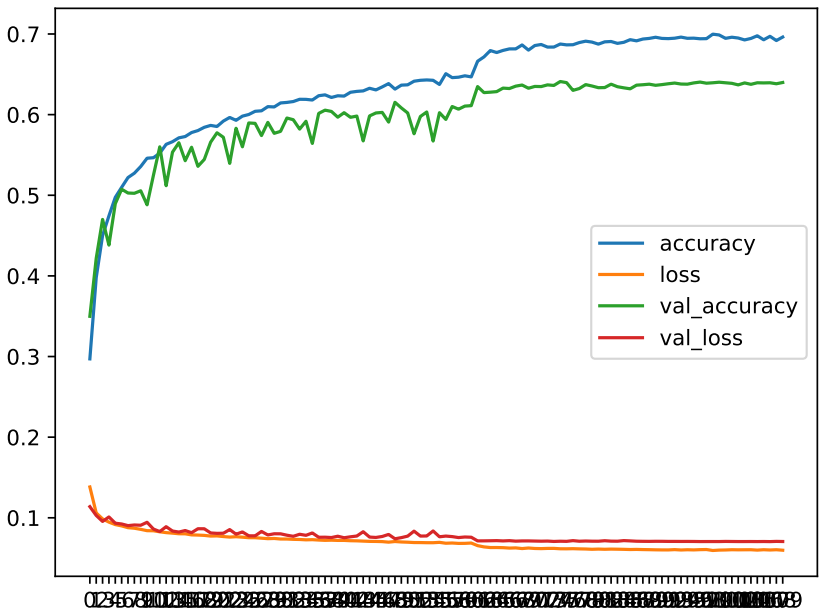

Figure 5.10: Training results using 28.273 images from the FER-2013 emotion dataset

Table 5.25: The precisin results against the CK+ dataset for the ninth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.84	45
Fear	0.39	25
Happy	0.93	69
Sad	0.45	28
Surprise	0.95	83
Neutral	0.00	1
Weighted Average	0.81	251

Table 5.26: The precisin results against the FER-2013 Public Test dataset for the ninth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.58	467
Fear	0.49	496
Happy	0.84	895
Sad	0.53	653
Surprise	0.76	415
Neutral	0.52	607
Weighted Average	0.64	3533

Table 5.27: The precisin results against the FER-2013 Private Test dataset for the ninth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.56	467
Fear	0.49	496
Happy	0.87	895
Sad	0.50	653
Surprise	0.74	415
Neutral	0.58	607
Weighted Average	0.64	3533

Analyzing the current situation:

There seems to be a slight improvement on the validation accuracy against FER-2013 from 64% to 65%, compared to our fifth and most successful trial. On the the CK+ dataset as well, from 79% to 81%, however, we decide to build the Confusion Matrix for the dataset we trained, on FER-2013, just to see if there are new confusions around.

Table 5.28: The confusion matrix for FER-2013 Public Test dataset for the ninth training trial

Emotions	**Angry**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Angry	**0.59**	0.09	0.04	0.11	0.03	0.13
Fear	0.11	**0.38**	0.04	0.24	0.08	0.15
Happy	0.02	0.02	**0.84**	0.02	0.02	0.07
Sad	0.10	0.11	0.04	**0.52**	0.02	0.21
Surprise	0.05	0.08	0.05	0.03	**0.77**	0.02
Neutral	0.06	0.05	0.09	0.16	0.02	**0.62**

Table 5.29: The confusion matrix for FER-2013 Private Test dataset for the ninth training trial

Emotions	**Angry**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Angry	**0.59**	0.10	0.02	0.15	0.02	0.12
Fear	0.15	**0.38**	0.03	0.20	0.11	0.13
Happy	0.04	0.02	**0.86**	0.03	0.02	0.04
Sad	0.12	0.09	0.04	**0.52**	0.02	0.22
Surprise	0.03	0.13	0.06	0.01	**0.73**	0.04
Neutral	0.04	0.05	0.05	0.16	0.02	**0.67**

From the confusion matrixes we observe that the only medium-high confusion left is between Fear and Sad. We continue with our tenth trial, and that means, training the on augmented FER-2013 dataset, but with the emotion Disgust removed, like here.

The tenth trial - augmented dataset without the emotion "Disgust":

- **Dataset**: Augmented FER-2013 x6, with the emotion "Disgust" removed, 169.638 images, split into 80% training and 20% validation sets. Each of the below points describes an operation made for 28.273 images from the FER-2013 dataset using a python library called imagaug, [5]:
  - **Original**: 28.273 images from the original FER-2013 dataset.
  - **Additive Gaussian Noise**: with a scale of 0.07*255.
  - **Multiply**: with random values between 0.25 and 1.50.
  - **Salt and Pepper**: with a percentage value of 0.03.
  - **Gaussian Blur**: with a value of 0.50.
  - **Clouds**: for each image a cloud like mask was put over.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 6] tensor containing the percentages for each label.
- **Loss function**: Mean Square Error.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a training validation accuracy of 68%, and a 78% accuracy on the CK+ dataset.

Table 5.30: The precisin results against the CK+ dataset for the tenth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.67	45
Fear	0.35	25
Happy	0.93	69
Sad	0.44	28
Surprise	0.97	83
Neutral	0.00	1
Weighted Average	0.78	251

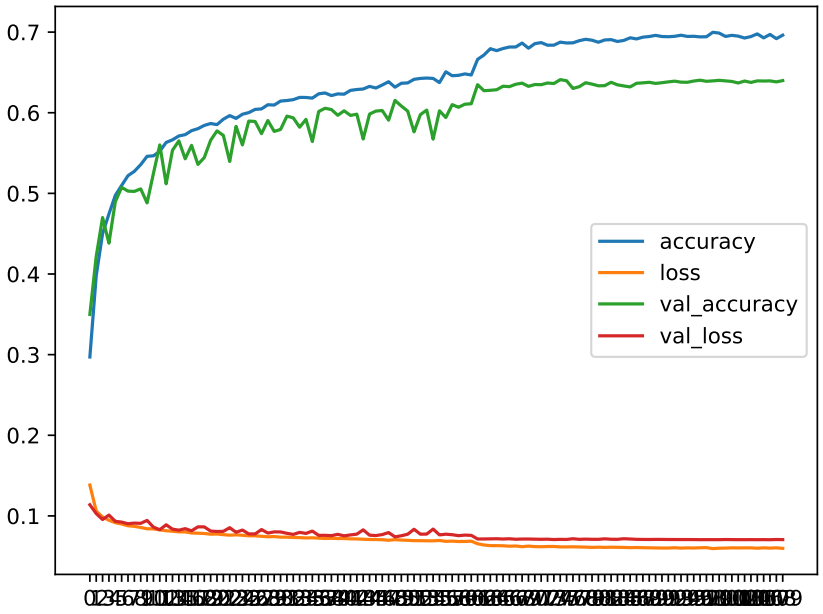

Figure 5.11: Training results using 169.638 images from the FER-2013 emotion dataset

Table 5.31: The precisin results against the FER-2013 Public Test dataset for the tenth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.58	467
Fear	0.52	496
Happy	0.85	895
Sad	0.57	653
Surprise	0.76	415
Neutral	0.58	607
Weighted Average	0.66	3533

Table 5.32: The precisin results against the FER-2013 Private Test dataset for the tenth training trial

Emotion	**Precision**	**No. of pictures**
Angry	0.59	467
Fear	0.51	496
Happy	0.85	895
Sad	0.49	653
Surprise	0.76	415
Neutral	0.63	607
Weighted Average	0.65	3533

Analyzing the current situation:

There seems to be a slight improvement on the validation accuracy against the ninth trial from 65% to 68%. On the the CK+ dataset as however, there is a slight decline from 81% to 78%. We decide to build the Confusion Matrix for the dataset we trained, on FER-2013 without the emotion Disgust, just to see the situation of confusions.

Table 5.33: The confusion matrix for FER-2013 Public Test dataset for the tenth training trial

Emotions	**Angry**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Angry	**0.63**	0.11	0.03	0.11	0.03	0.09
Fear	0.12	**0.45**	0.03	0.21	0.08	0.11
Happy	0.02	0.02	**0.87**	0.01	0.02	0.05
Sad	0.11	0.11	0.05	**0.55**	0.02	0.16
Surprise	0.03	0.09	0.04	0.03	**0.78**	0.02
Neutral	0.08	0.06	0.10	0.15	0.02	**0.59**

Table 5.34: The confusion matrix for FER-2013 Private Test dataset for the tenth training trial

Emotions	**Angry**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Angry	**0.61**	0.09	0.03	0.16	0.02	0.10
Fear	0.14	**0.45**	0.04	0.18	0.09	0.10
Happy	0.04	0.02	**0.87**	0.03	0.02	0.03
Sad	0.10	0.13	0.06	**0.51**	0.02	0.17
Surprise	0.03	0.12	0.05	0.02	**0.76**	0.01
Neutral	0.05	0.06	0.07	0.17	0.02	**0.63**

From the confusion matrixes we observe that the only medium confusion left is between Fear and Sad. Further investigation on this matter would imply a train for just Fear and Sad to see if the neural network can learn the difference between them or not.

The eleventh trial - FER-2013 Sad and Fear:

- **Dataset**: Sad and Fear emotions from FER-2013 dataset, 9.927 images, split into 80% training and 20% validation.
- **Input size**: [1, 48, 48, 1] grayscale pixel images. Each pixel, a value between [0, 255].
- **Output size**: [1, 2] tensor containing the percentages for each label.
- **Loss function**: Mean Square Error.
- **Optimizer**: Adam.
- **Learning rate**: 0.1.
- **Batch size**: 32.
- **Epochs**: 110.
- **Training metric**: Accuracy.

The result was a training validation accuracy of 71%, and a 74% accuracy on the CK+ dataset.

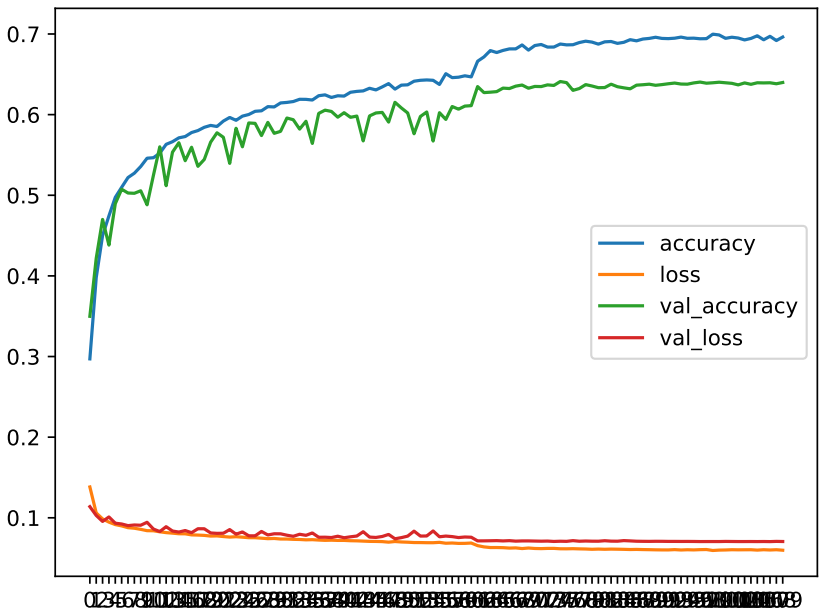

Figure 5.12: Training results using 9.927 sad and fear images from the FER-2013 emotion dataset

Table 5.35: The precisin results against the CK+ dataset for the ninth training trial

Emotion	**Precision**	**No. of pictures**
Fear	0.85	25
Sad	0.65	28
Weighted Average	0.74	53

Table 5.36: The precisin results against the FER-2013 Public Test dataset for the ninth eleventh trial

Emotion	**Precision**	**No. of pictures**
Fear	0.70	496
Sad	0.73	653
Weighted Average	0.72	1149

Table 5.37: The precisin results against the FER-2013 Private Test dataset for the eleventh training trial

Emotion	**Precision**	**No. of pictures**
Fear	0.73	496
Sad	0.72	653
Weighted Average	0.72	1149

Analyzing the current situation:

Considering we trained for just two emotions, the results indicate there is indeed quite a lot of confusion between these two emotions. We decide to build the Confusion Matrix to investigate this.

Table 5.38: The confusion matrix for FER-2013 Public Test dataset for the eleventh training trial

Emotions	**Fear**	**Sad**
Fear	**0.60**	0.40
Sad	0.20	**0.80**

Table 5.39: The confusion matrix for FER-2013 Private Test dataset for the eleventh training trial

Emotions	**Fear**	**Sad**
Fear	**0.65**	0.35
Sad	0.21	**0.79**

From the confusion matrixes the confusion between Fear and Sad is obvious. Since Sad is learned better than Fear, the next point of action could be removing the Fear emotion from the dataset. Other points for investigation could be training on different datasets and comparing the results with FER-2013.

Chapter 6

Acting Mirror

Coming back to our main purpose of studying the FER problem, we will take a look at the application we have developed. Acting Mirror takes a video stream from the camera, from a file on the machine it runs on, or from a youtube video, and processes the video stream drawing rectangles around the face of people in the stream and labeling them with one of 7 emotions: "angry", "disgust", "scared", "happy", "sad", "surprised", "neutral".

As illustrated in 6.1, the app works by reading a video stream, and passing that stream to a the haarscade frontal face detector. For each detected face, an intelligent alogirthm parses the pixels of the face in the format of a 48x48 grayscale pixel matrix, and labels the face with one of 7 emotions. In the end, the processed output video stream is shown to the user and also saved in a file called "output.avi".

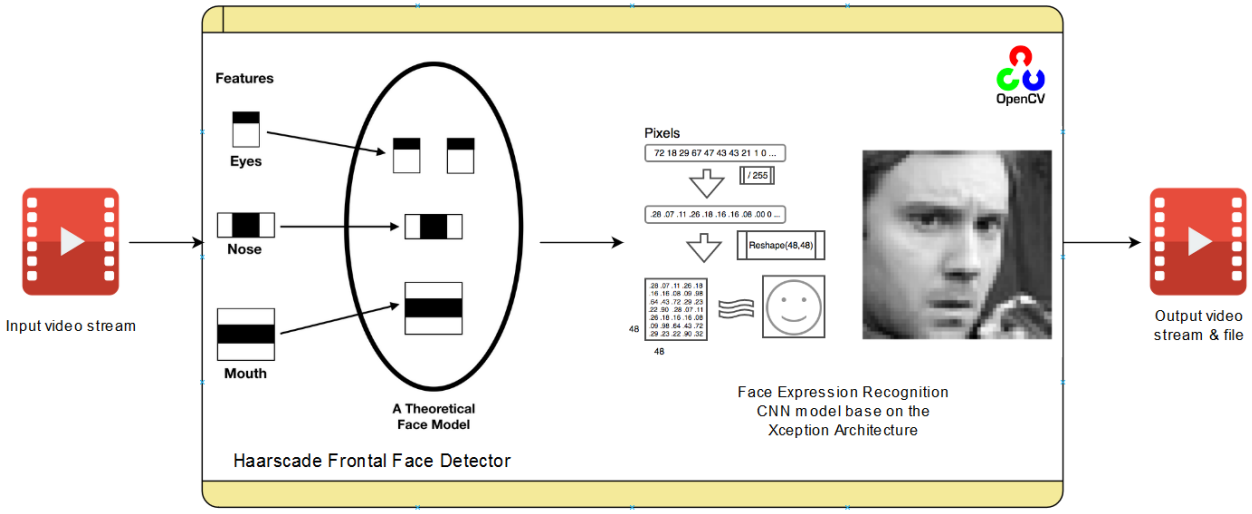

Figure 6.1: Acting Mirror Architecture

As illustrated in 6.2, Acting Mirror also presents a Graph with the percentage emotions over time

of person 0 (the biggest person/face in the video stream) and a real-time percentage of each frame's emotion. All the details and options of running Acting Mirror can be found at our GitHub repository at Acting Mirror - Research and Development.

Figure 6.2: Streaming the "Hotdog, not Hotdog" scene from Sillicon Valley Season 4

A few statistics about running Acting Mirror on our machine:

- **Machine description**: CPU i7-7500u and an NVIDIA Quadro M520 1GB.
- **FPS** running the whole Acting Mirror either with Keras or TFLite Model:
  - **0 People**: 14.6
  - **1 Person**: 11
  - **20 People**: 3.5
- **Average time** to classify one face on **Keras** model: 0.00587 seconds / face
- **Average time** to classify one face on **TFLite** model: 0.00374 seconds / face

Chapter 7

Conclusion and future work

7.1 State of the Art Comparison

During our research, we have focused a lot on finding a better set of independent variables that would help the model from [1] reach a better accuracy on the FER-2013 dataset. On each trial we have tested the model against the FER-2013 test sets (public and private) and against the CK+ dataset. The best results we got were on the fifth trial, with 64% accuracy on the FER-2013 dataset and 79% accuracy on the CK+ dataset. After augmenting the dataset, on the eight trial, we got an accuracy of 68% on the FER-2013 dataset, and just 71% accuracy on the CK+ dataset.

We decided to generate confusion matrixes in order to better analyze our situation, and what we found was that the FER-2013 dataset contains some confusions between emotions like Angry and Disgust, Sad and Fear, and between the aforementioned emotions and Neutral in smaller parts. We experimented with excluding the emotion Disgust from the dataset, but the improvements were not significant. We proposed 2 action points: either excluding the Fear emotion as well, which receives a big amount of confusion from Sad, or to train the model on a better dataset.

Comparing other state of the art results that trained on the CK+ dataset against our achievements looks like we need a lot of improvement to do in order to catch up.

Table 7.1: Comparison between State of the Art results and our results

State of the Art	**Deep Networks [6]**	**SVM [6]**	**CNN [2]**	**Our results - model [1]**
CK+ Dataset	48-96	31-50	93-99	71-79

7.2 Brief Ethical Discussion

There are a few problems we see with regards to the FER problem. We believe it may be ok to use it, but just as long as it is not intrusive and misused.

Understanding that the current approach to the problem is not perfect is also important with regards to using these tools. For instance, when a dataset labels a picture as "Happy", is that person really happy? Or are they just smiling, laughing or having an uncontrollable hysterical laughther? Not all laughter or smile should be associated with happiness, which is a state that can be manifested through a neutral, poker face, as well.

The context where FER would be applied should be an accepted context by all its users and only manifested for as long as accepted by the people and only for as long as it brings a positive value to the people. In figure 7.1 we propose a mind-map which describes 5 such applications.

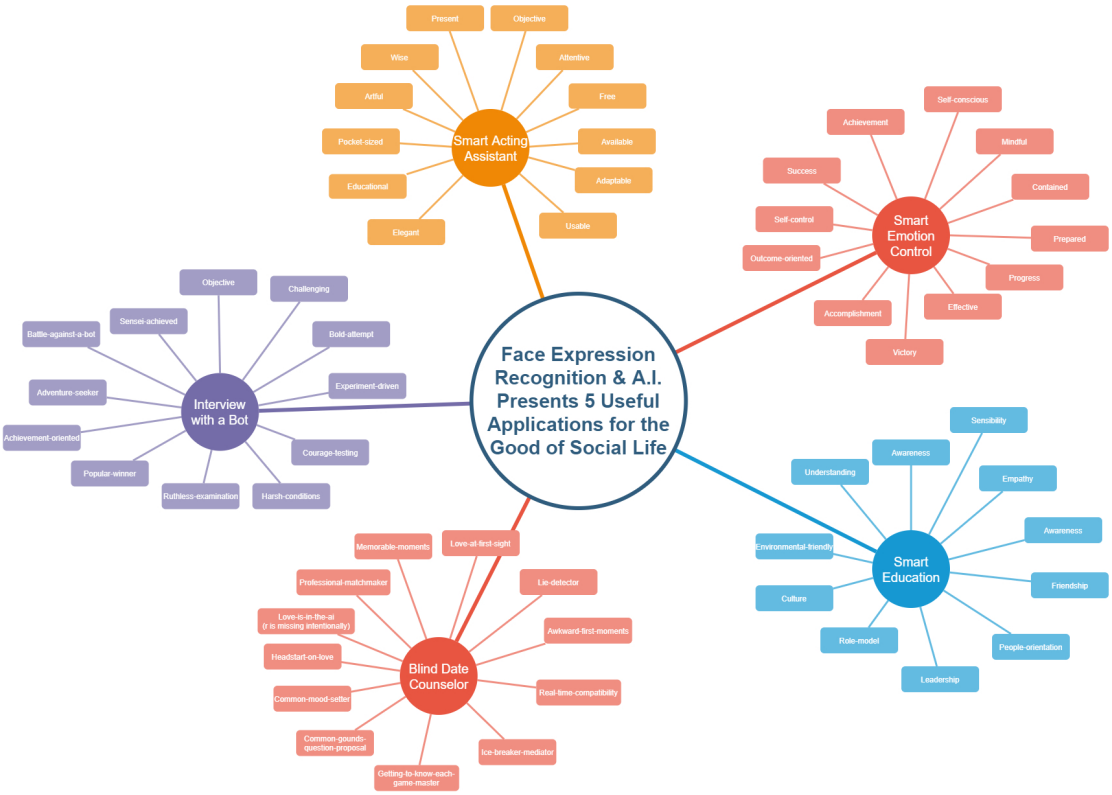

Figure 7.1: Applications for FER - MindMap

7.3 Acting Mirror and Future Work

Our practical contribution, Acting Mirorr is an applicaiton which takes a video stream from the camera, from a file on the machine it runs on, or from a youtube video, and processes the video stream drawing rectangles around the face of people in the stream and labeling them with one of 7 emotions: "angry", "disgust", "scared", "happy", "sad", "surprised", "neutral".
Further development of Acting Mirror could mean bringing it to mobile devices, optimizing the stream of data in the application, or adding new features to it, such as the ability to parse a script and give real-time audio feedback to actors about their performance or tips that may help them further increase their acting skills.

Other future work involving the FER problem could be creating and validating good face expression emotion datasets. One thing we learned was to search for confusion matrixes of the dataset we intend to use and study on before actually starting the work. It would have probably saved us a lot of trial and error time or enable us to achieve greater results.

Studying the ethical issues related to the FER problem is something we need now more than ever. With tools that are already developed, or even better ones in development, is only a matter of times before we, as a human race, are in a situation of having tools we do not properly understand their intended use or value. Making sure we are prepared and education on how and when to use such tools is of utmost importance.

In conclusion, we believe we are on the right track when it comes to developing tools that make use of intelligent algorithms, such as those capable of learning the FER problem, its just very important that we also become prepared and educated on how to use them. Experimenting, however, is part of the journey, which is why we propose the 5 applications from the MindMap at 7.1 as a next stepping stone in applied FER.

Bibliography

[1] Octavio Arriaga, Paul G. Ploger, and Matias Valdenegro. Real-time convolutional neural networks for emotion and gender classification. 2017.

[2] Peter Burkert, Felix Trier, Muhammad Zeshan Afzal, Andreas Dengel, and Marcus Liwicki. Dexpression: Deep convolutional neural network for expression recognition. *German Research Center for Artificial Intelligence (DFKI)*, 2016.

[3] Catalin Caleanu. Face expression recognition: A brief overview of the last decade. *SACI 2013 - 8th IEEE International Symposium on Applied Computational Intelligence and Informatics, Proceedings*, pages 157–161, 2013.

[4] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. *CoRR*, 2016.

[5] Alexander Jung. imgaug - a python library.

[6] Najmeh Samadiani, Guangyan Huang, Borui Cai, Wei Luo, Chi-Hung Chi, Yong Xiang, and Jing He. A review on automatic facial expression recognition systems assisted by multimodal sensor data. *Multidisciplinary Digital Publishing Institute (MDPI)*, 2019.

[7] Pawel Tarnowski, Marcin Kolodziej, Andrzej Majkowski, and Remigiusz J. Rak. Emotion recognition using facial expressions. 2017.