INSTITUTO TECNOLÓGICO DE BUENOS AIRES – ITBA

ESCUELA DE INGENIERÍA Y GESTIÓN

Detección Automática de Expresiones Faciales por medio de Imágenes

AUTOR/ES: Casagrande, Lucas (Leg. N° 55302)
Kuyumciyan, Nicolás (Leg. N° 55165)

DOCENTE/S TITULAR/ES O TUTOR/ES: Gambini, Juliana

TRABAJO FINAL PRESENTADO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN INFORMÁTICA

Lugar: Lavardén 315, C1437 CABA, Argentina
Fecha: 05 de Julio de 2019

Automated detection of facial expressions using image analysis

Lucas Casagrande, Nicolás Kuyumciyan

July 6, 2019

Contents

Automated detection of facial expresions using image analysis

Authors: Lucas Casagrande and Nicolás Kuyumciyan

Tutor: Dr. María Juliana Gambini

Instituto Tecnológico Buenos Aires (ITBA)

1 Abstract

1.1 English

The broad range of applications for automatic expression detection sparks the need for a robust and effective implementation. In this paper, an exposition of the existing methods most frequently used for this purpose is done, and an analysis of their performance is carried out. These include both feature detection methods such as Gabor filters and Histograms of Gradients as well as classifiers based in neural networks. Existing data sets consisting of images of persons faces with a labeled expression are used for training and testing purposes. A success rate of 87.6% is achieved when classifying images with up to four different expressions.

1.2 Español

La amplia gama de aplicaciones para la detección automática de expresiones faciales genera la necesidad de implementar un sistema rápido y robusto. En este trabajo se realiza una exposición de los métodos existentes más frequentemente usados y se lleva a cabo un análisis de su rendimiento. Éstos incluyen tanto métodos para la detección de características como clasificadores basados en redes neuronales. Se utilizan bases de datos existentes de imágenes de personas, con una expresión etiquetada, para entrenar los métodos de clasificación y para evaluar la efectividad de los mismos. Se alcanza un éxito del 87.6% cuando se clasifican hasta cuatro expresiones distintas.

2 Introduction

Facial expressions are one of the fundamental characteristics which humans possess to be able to communicate their emotions and feelings. There is a growing need to develop fast and robust methods of automating the capturing and interpretation of these expressions. This need arises due to the imminent advances in robotic automation in the workplace, and the positive impacts it has [25], coupled with the wide range of use cases facial expression recognition has.

Even though for the human eyes and brain it is natural, intuitive and immediate to detect and interpret these expressions, there is a significant grade of complexity for computers to achieve these results.

It is important to distinguish between recognition of expressions and recognition of emotions when the medium are images. Facial expressions recognition deals with the classification of facial movements and deformations into abstract classes such as happiness, sadness, surprise and others. This is based exclusively on visual information of a single moment in time and hence cannot interpret other factors such as posture, voice tone, gestures, eye contact and others which are essential to determining the emotion a person is experiencing [5].

2.1 Motivations

Automatic detection of facial expressions has a diverse set of applications. These include but are not limited to the following:

1. **Monitor a driver's attention**

   It is proven that 20% of all automotive accidents involve a fatigued driver, most commonly caused by a lack of sleep [6]. An automated expression detection system integrated with the vehicle could monitor in a non-intrusive manner the state of fatigue in the driver [11]. Figure 1 shows an example of what such a system could look like.

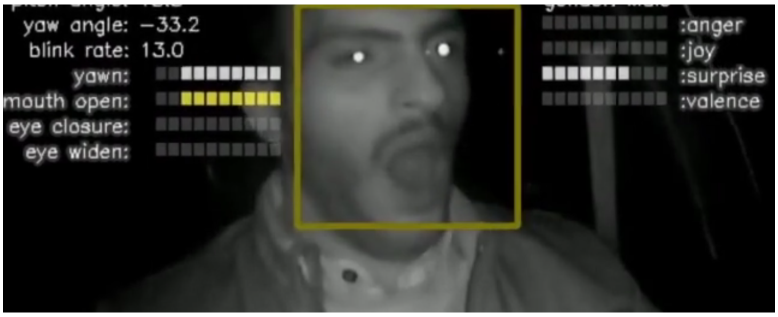


Figure 1: Example of fatigue detection in a driver.

   Determining the state of fatigue of a person is complex and subjective. A single picture is not be sufficient to determine if someone is tired, instead the implemented system would need to analyze the state of fatigue over time make its decision on whether or not the driver is tired. The action to take if the system decides the driver is tired could be to warn the driver, limit the maximum speed or, in the case of commercial drivers, warn the company.

2. **Measure clients satisfaction**

   Facial expressions play an important role in all social interactions. Hence, companies have begun to put their focus on currently available detection systems [12]. Studies have shown that when people communicate, the emotion transmitted is represented only a 7% by language used, 38% by the tone of voice and 55% by facial expressions [19].

In current times, more and more companies are focusing in the client as a key piece in their success, trying to gain their loyalty by providing solutions to all the problems a customer may face. Being able to detect in real time when a customer is having a bad experience, and acting upon it before the customer complains is a game-changer.

3. **Virtual reality video games**

   In the world of video games the industry tendency seems to be a shift towards virtual reality games [14]. Technology giants such as *Facebook* have already unveiled the first facial expression recognition systems for such a purpose [24]. In recent years investigations in face expression recognition have been carried out [1] with the express purpose of enhancing the player experience in virtual reality games by making them more realistic.

   The use of facial expression recognition allows for an extra layer of immersion by more accurately visually representing the player. Given virtual reality already provides information on the users gestures and tone of voice, combining these with facial expression recognition can provide a more precise approximation to the persons emotion than either of them alone. With this information new functionalities can be integrated to games such as acclimating the virtual world to the players mood or adapting interactions with computer controlled characters so that their reactions are not only a result of the players actions but of their demeanour as well.

4. **Personalized music recommendation**

   The digital music industry is growing each year more, having over half of the market share of all the musical industry (54%) [22]. Given its a highly competitive market, music streaming providers need adequate unique selling points one of which is personalized music recommendations. Current methods for the selection of this music are separated into three categories: content, metadata or emotion [4]. This is not necessarily restricted to using a single method at a time, a combination of them is the best approach.

   Content based selection is focused on previous listening patterns of the user. Metadata based selection depends on information collected about the user outside the listening patterns such as nationality, location, age and others.

   Emotion based selection is founded on the user's emotional state, recommending appropriate songs depending on their mood. One of the big disadvantage of this method over content and metadata based selection is the fact that its not as accessible to get the required information to make an educated emotion based decision.

5. **Security**

   The key component of security is prevention, stopping the crime from happening. Expression recognition through image analysis of security cameras provides a way of detecting the crimes as they are about to happen by detecting someones intention to steal before it happens by noticing signs of nervousness or erratic behaviour [20].

2.2 Structure

In the next section 3 and overview is done of the current state of the art in expression detection through image analysis. In section 4 a detailed explanation of the methods used in the implementation will be carried out. Section 5 lists the image data sets used and their characteristics, highlighting strengths and weaknesses of each one. In section 6 a thorough exposition is carried out on the results of implemented program, with subsections for the implementation of each separate method, while in section 7 a synthesis of the conclusions are made along with potential areas to explore further.

3 State of the art

In this section a description of the current state of the art methods in regards to facial expression recognition through image analysis will be made.

3.1 Process for facial expression recognition

Automatically arriving at the conclusion that in an image there is a person that has a certain expression is a process with several stages. Figure 2 graphically outlines this process, which includes image pre-processing to eliminate noise, the detection of the face's location in the image, gathering facial features and the classification of these features to one or more expressions [9, 13].

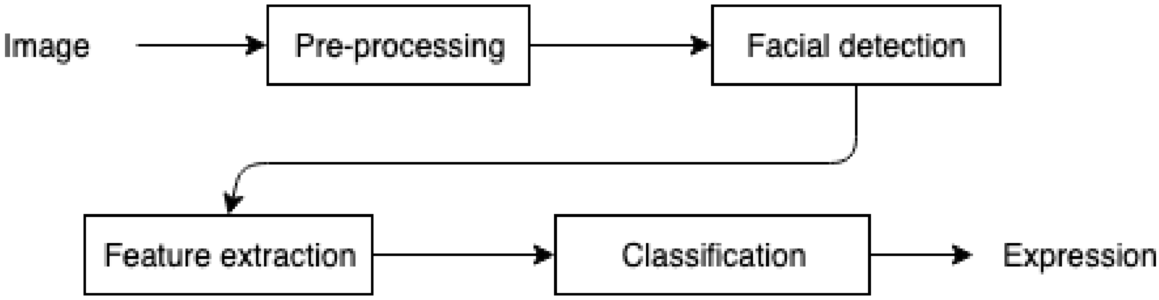


Figure 2: Stages in the process of facial expression recognition in images.

All stages of the process can be divided into two encompassing groups: the detection and extraction of features and the classification of such features into a specific expression.

3.1.1 Methods based on features detection

Prior to being able to assign a certain expression to the person in an image it is necessary to obtain the features which will be evaluated to determine which expression is represented.

One of the most utilized techniques is to employ Gabor filters [16]. Gabor filters distinguish themselves for their qualities as noise resistant and for being invariant to changes in lighting, rotation, scale and translation. Gabor's 2D filters are a Gaussian function modulated by a sinusoidal plane wave given by the equation (1).

$$\begin{aligned} G(x,y) &= \frac{f^2}{\pi\gamma\eta} \exp(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}) \exp(i2\pi f x' + \phi) \\ x' &= x\cos(\theta) + y\sin(\theta) \\ y' &= -x\sin(\theta) + y\cos(\theta) \end{aligned} \tag{1}$$

Where f is the frequency of the sinusoidal factor (inverse of the wave length), θ is the orientation of the normal to the parallel bands, ϕ is the phase offset, σ the standard deviation and γ the spatial aspect ratio. Figure 3 shows the graphical representation for Gabor filters with different orientations and frequencies.

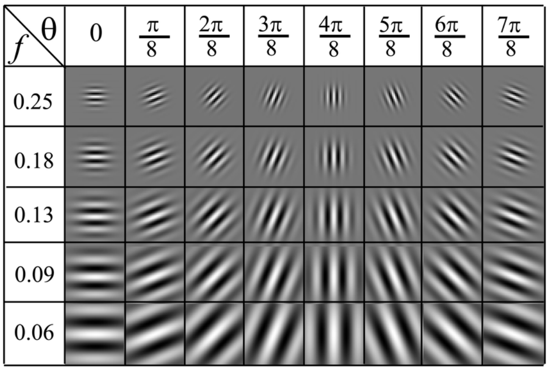


Figure 3: Gabor filters generated with different combinations of θ (in radians), f (in Hz) and $\phi = 0$.

The implementation proposed in the article [8] uses several filters with different orientations and scale, with the results of applying these filters then downsampled (lowering of resolution). Given the features vector is still of a significant size (in the example there are still 36000 values for images of dimensions 120x120), a dimensional reduction is carried out using the linear discriminant analysis (LDA) [7], arriving at a feature vector of just 199 elements.

3.1.2 Methods based on classification

Classification consists in assigning one or more expressions to the person detected in the image derived from analyzing the extracted features. Some of the most common classification methods are SVM (Support Vector Machine), clustering algorithms and the use of neural networks.

Classification by use of SVM's is a supervised type of classification and consists of creating an N-dimensional vector with the N features extracted from the image. The vector machine is then trained to partition the space in k different categories (in this case study each category is an expression). The hyper-space of each partition will contain all points which belong to that expression. To classify an image it is simply a matter of deriving its vector from the features and assign it to the expression associated with the hyper-space to which it belongs [2].

Grouping or clustering algorithms build a predetermined number of data groups by calculating the distance between the data points [23]. Amongst the clustering algorithms the most common ones are K-means, K-medoids and CLARANS.

Neural networks used as classifiers can be very robust [17] but require a significant amount of data for training and are more difficult to implement.

4 Description of used methods

In this section the methods used are described. The methods are grouped into two subsections, the methods for feature detection and the methods for classifying the output of feature detection.

4.1 Feature detection

The aim of feature detection methods is to reduce the input from being the whole set of pixels in the image to a smaller subset of values which are representative of the expression it is trying to convey. Most methods achieve this by outputting the information on borders which are representative of changes in a persons face.

4.1.1 Gabor Filters

As with all image filters, they consist of a two dimensional matrix and are applied by performing a convolution with the original image. Gabor filters are lineal filters that weigh both the orientation as well as any frequency pattern in an image. The result of applying a Gabor filter to an image is a new image where borders perpendicular to the orientation of the filter predominate as seen on Figure 4.

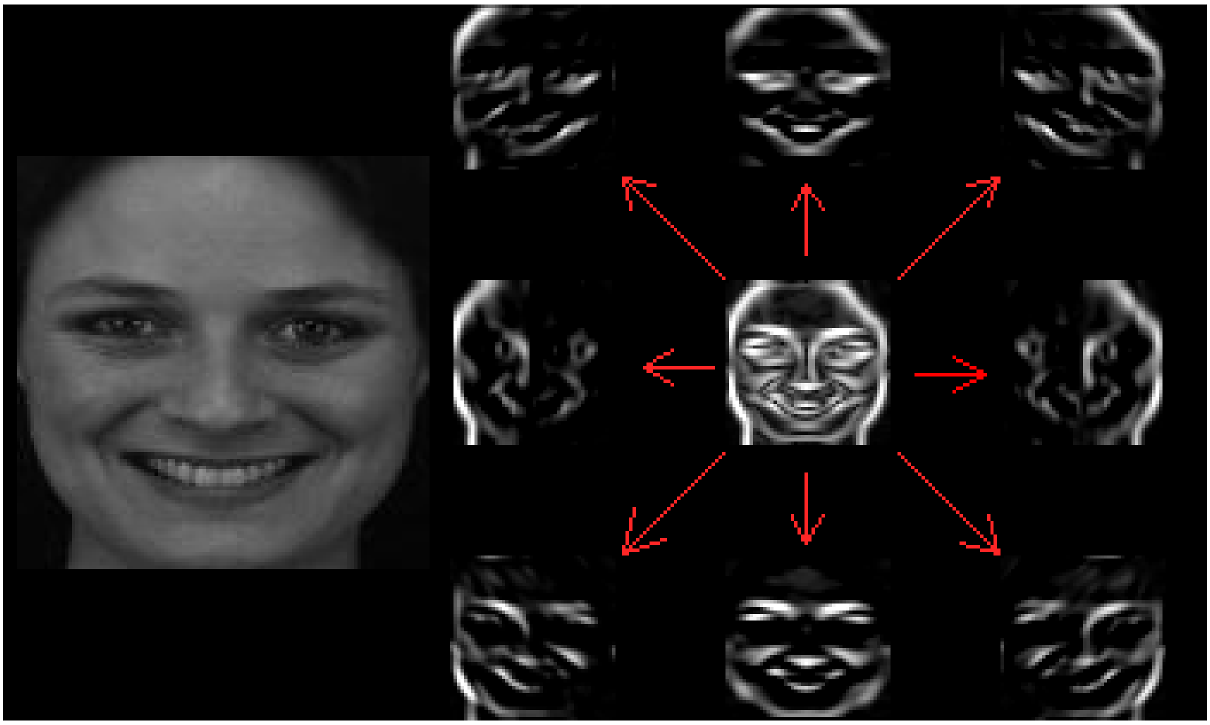


Figure 4: A face and the result of applying a Gabor filter with different orientations, and the superposition of all the filters in the middle

Gabor's convolution matrix is based on the Gaussian kernel and is defined by Equation 1.

These filters distinguish themselves for being resistant to noise and are invariant to changes in illumination, rotations, scale and translation.

Gabor filters detect borders or pronounced changes in levels of grey in an image. Therefore, when applied to the image of a persons face the borders can be considered a distinctive feature of the person and can be used as the input to a classification method.

Several authors [3, 18] propose that Gabor filters are a representative model of the receptive fields of simple cells in the cerebral cortex. Due to this, using them could be a first step towards being able to interpret images in a similar way in which the human brain does it.

4.1.2 Histogram of oriented gradients (HOG)

The histogram of oriented gradients method is performed by following these steps:

- Calculate the gradients en each pixel of the image by using the convolution matrix $[1, 0, -1]$ and its transposed version.
- The gradient vectors are then discretized in n directions. This value is often called orientations.
- the vectors are grouped in cells of size $k \times k$ and then a histogram is created for every cell
- To account for changes in illumination and contrast, the gradient strengths must be locally normalized, which requires grouping the cells together into larger, spatially connected blocks.
- The HOG descriptor is then the concatenated vector of the components of the normalized cell histograms from all of the block regions.

The parameters are:

- Size of the window to take into consideration. It has to be constant across all histograms. In this case a $48x48$ window is, picked which matches the size of the image.
- Size of the cell. This is the amount of pixels to be grouped for each bucket in the histogram
- Size of the Block. This has to be a multiple of the size of the cell given each block is formed by groups of cells
- Block step. This is the amount of pixels between the start of a block and the next. It does not necessarily need to match the size of the block, if it is smaller then some pixels will be in more than one block at the same time.
- Number of orientations in the histogram. Default is 9.
- (Optional) Can specify if the gradients are signed or not.

Figure 5 shows the face of a happy person and the result of applying the histogram of gradients method. White pixels represent pixels in the original image where the gradients magnitude is greatest.

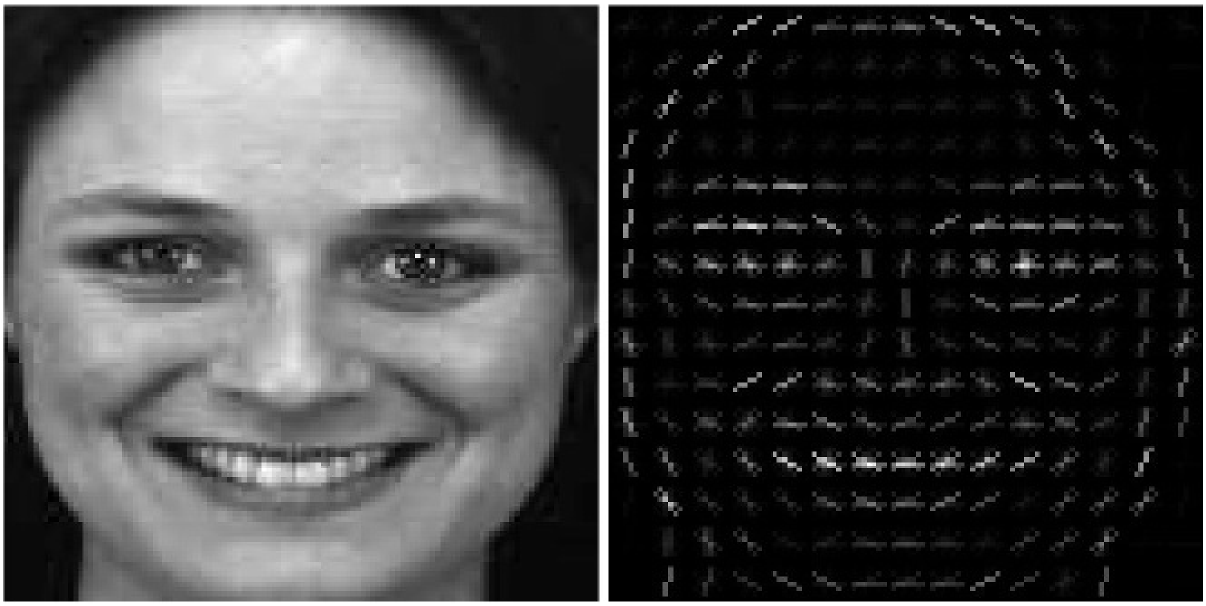


Figure 5: Original image (left) and the result of applying the HOG method (right).

4.2 Classification

Classification algorithms use as an input the reduced feature vector produced by feature detection algorithms and output an expression.

4.2.1 Feed-forward Neural Networks (FFNN)

Artificial neural networks (ANN) are computing systems that are inspired by, but not necessarily identical to, the biological neural networks that constitute animal brains. Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules.

An ANN is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal from one artificial neuron to another. An artificial neuron that receives a signal can process it and then signal additional artificial neurons connected to it.

Feed-forward neural networks are a subset of all the different neural networks. They are characterized by having all the neurons of a layer fully connected to all the neurons of the next layer as it can be seen on Figure 6. This makes that all the information of a given layer is passed to the next layer which transform that information. This goes all the way from the input layer whose values are the data that is being fed to the output layer. The output layer will hold the values which determine into which category the input is classified. The layers in this neural network are often referred to as Dense Layers.

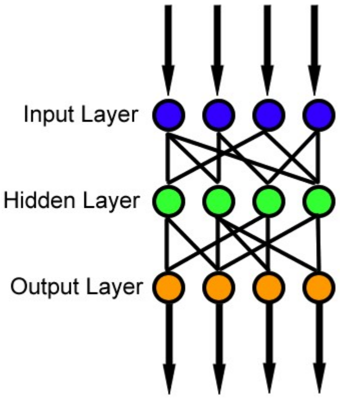


Figure 6: An example of a Feedfoward neural network.

4.2.2 Convolutional Neural Networks (CNN)

A Convolutional Neural Network (CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, convolutional neural networks have the ability to learn these filters/characteristics.

A CNN differs from a FFNN in that each layer of the CNN is not fully connected to next one. A CNN performs two basic operations, a convolution and a sub-sampling.

Convolution

Convolution is an operation in which a kernel (matrix of values) is used to modify another matrix by weighing the values of the neighbouring elements. The result of this operation is then stored on a new matrix, an example of this can be seen on Figure 7.

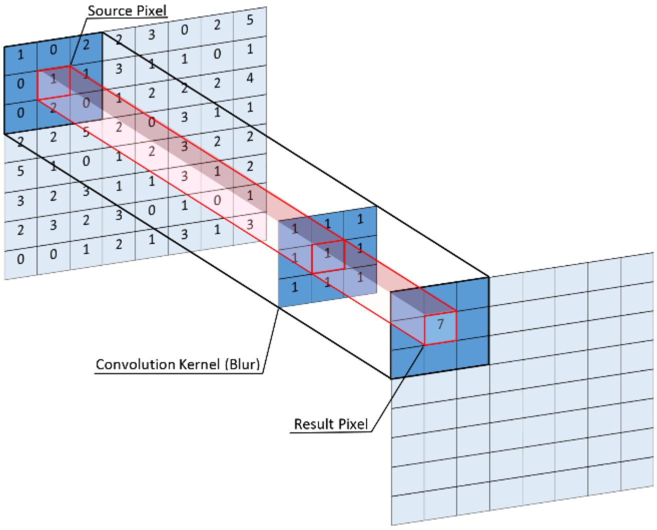
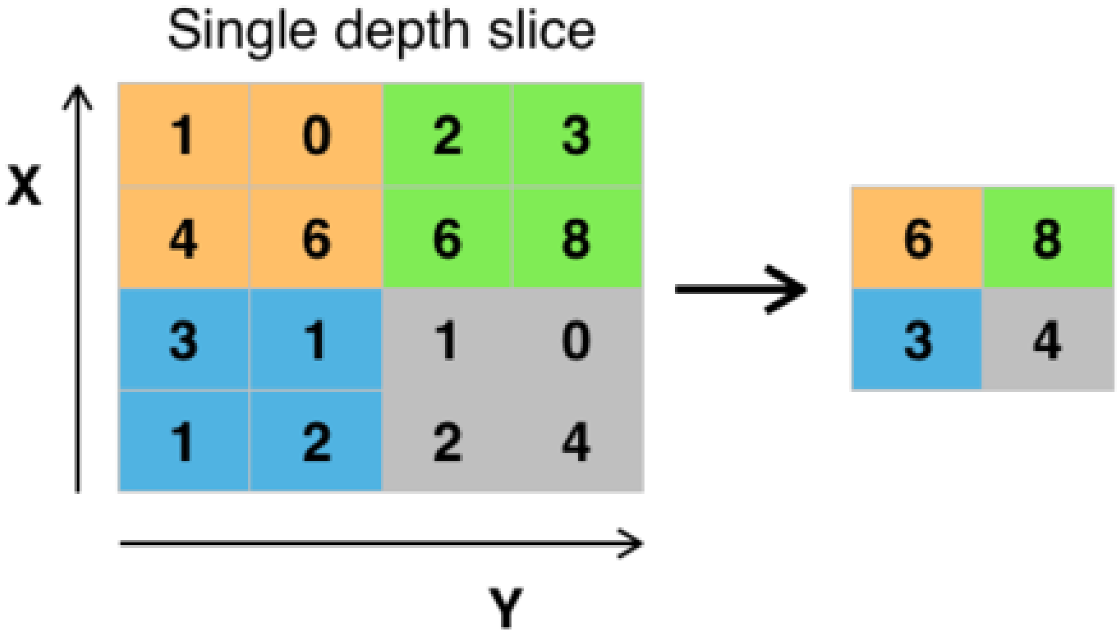


Figure 7: An example of a convolution.

The objective of the Convolution Operation is to extract the high-level features such as edges, from the input image. Convolutional neural networks need not be limited to only one Convolutional Layer. Conventionally, the first ConvLayer is responsible for capturing the Low-Level features such as edges, color, gradient orientation, etc. With added layers the architecture adapts to the High-Level features as well, resulting in a network which has the understanding of images as a whole, similar to how the human brain interprets them.

Pooling

Another common operation in a CNN is the subsampling, this is done by the Pooling Layer. This is done to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training of the model. An example of a CNN that uses subsampling can be seen on Figure 9.



1	0	2	3
4	6	6	8
3	1	1	0
1	2	2	4

6	8
3	4

Figure 8: An example of a max pooling layer and its result.

There are two types of Pooling: Max Pooling and Average Pooling. Max Pooling returns the maximum value from the portion of the image covered by the Kernel as seen on Figure 8. On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel.

Max Pooling also acts as a Noise Suppressant. It discards the noisy activations altogether and also performs dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, it can be said that Max Pooling performs a lot better than Average Pooling.

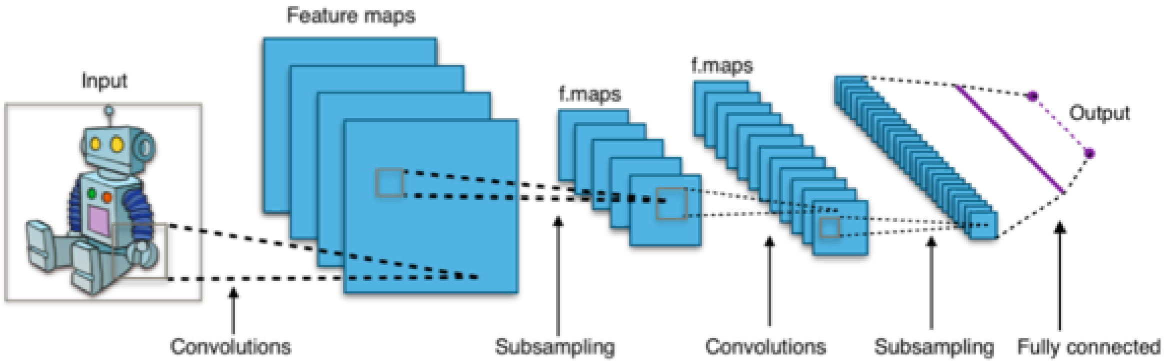


Figure 9: An example of a convolutional neural network.

After a certain amount of Convolutional and Pooling Layers are defined, it is necessary to switch to a fully connected layer, in order for the Neural Network to be able to return a result. This is done by flattening the last layer, and then creating some more layers. A diagram of a CNN with all the layers can be seen in Figure 9

Batch Normalization

Other types of layers exist. One of them is called batch normalization [10]. Batch normalization reduces the amount by which the hidden unit values shift around (covariance shift). To increase the stability of a neural network, batch normalization normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation.

Data Augmentation

Another technique used in tests is called Data Augmentation. Data Augmentation is a technique where an image from the data set is picked and applied transformations to end up with a similar but slightly different image. The operations may include, but are not limited to rotation, translation, shearing, flipping or zooming. This increases the size of the training data, and contributes towards avoiding over-fitting.

5 Data sets

When it comes to supervised algorithms having a robust data set is essential to the overall performance of the program. It is important because it is the single source of truth from which the algorithm will derive all its inputs. Hence a data set with wrong labels, such as having an image of a person which is clearly sad labeled as happy, will have an impact on how well it will later on be classified. The more discrepancies in the data set, the worse the classifier will perform. Another issue with the data set could be that, even if everything is correctly labeled, there are some cases which are not thoroughly covered or there is no data at all. For example, if in the data set there are only images where all the subjects are female then the algorithm would struggle when it gets asked to classify the picture of a male person. Following this same reasoning not only it would be important to have diversity in gender but also in ethnicity, ages and anything else that would impact the image of a persons face such as wearing glasses, having facial tattoos, piercings, etc.

5.1 KDEF

One of the data sets used is the Karolinska Directed Emotional Faces (KDEF) [15]. It consists of 4900 images of human facial expressions displaying seven different emotional expressions: angry, afraid, disgusted, neutral, sad, surprised and happy. A sample of images from the KDEF data set can be seen on Figure 10.

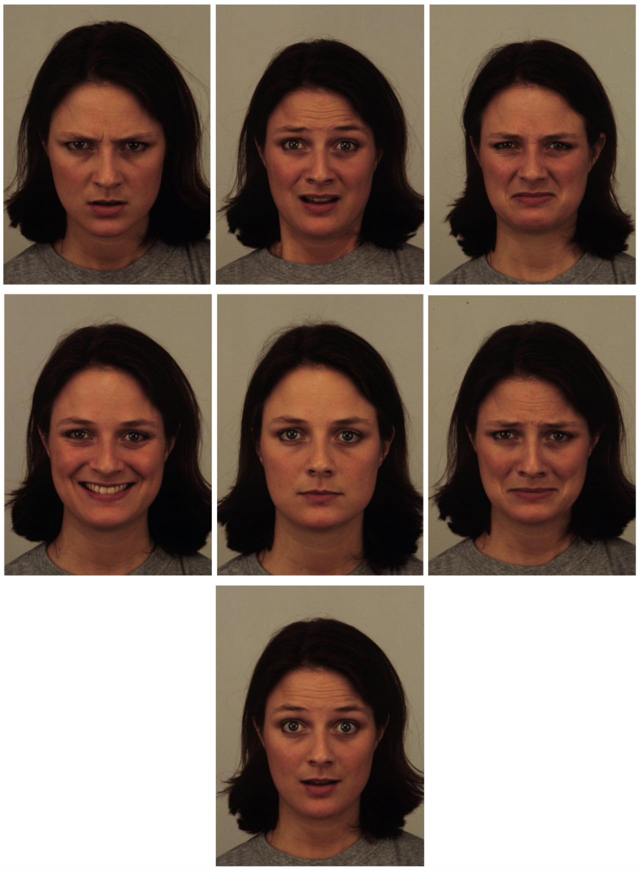


Figure 10: Images from the KDEF data set depicting the same person captured with all seven different expressions. Anger, fear, disgust, happiness, neutral, sadness and surprise respectively.

There are seventy different subjects, thirty five male and thirty five female with every subject posing for each of the seven expressions. Each expression is pictured from the following different angles: full left profile, half left profile, straight, half right profile, full right profile. There are two sessions so each subject poses for a particular expression twice.

The selection criteria for subjects consists of people aged between twenty and thirty years of age with no beards, mustaches, earrings or eyeglasses, and preferably no visible make-up during photo-session. Subjects are wearing monotone grey T-shirts and are seated three meters from the camera, with the absolute distance adapted for each subject so that the eyes, mouth and other distinctive features lined up in all images.

Only images on the straight frontal position are used for this implementation so a subset of just under one thousand images is used.

5.2 FACES

Another data set used is FACES [21]. This data set consists of a total of 2052 images from 171 different subjects portraying six separate types of facial expressions. The expressions are neutrality, sadness, disgust, fear, anger, and happiness and each subject poses for each expression twice from a straight frontal perspective. An example of the pictures taken can be seen on Figure 11.

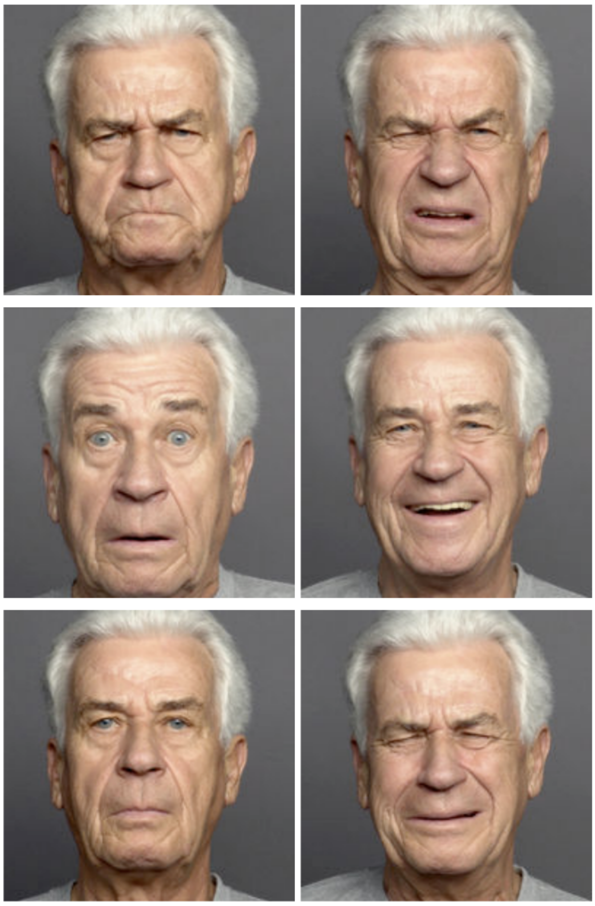


Figure 11: Images from the FACES data set depicting the same person captured with all six available expressions. Anger, disgust, fear, happiness, neutral and sadness respectively.

Contrary to the KDEF data set, FACES consists of a more diverse subject population in terms of age. They are classified into three separate groups of 58 young, 56 middle aged and 57 old subjects with all groups having both males and females.

5.3 FER2013

This is the biggest data set used, consisting of 28,709 examples. Because of the magnitude of the data set, which is collected from different sources on the internet, some strange samples may be part of this data set as seen on Figure 12.

The data consists of 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is approximately centered and occupies about the same amount of space in each image.

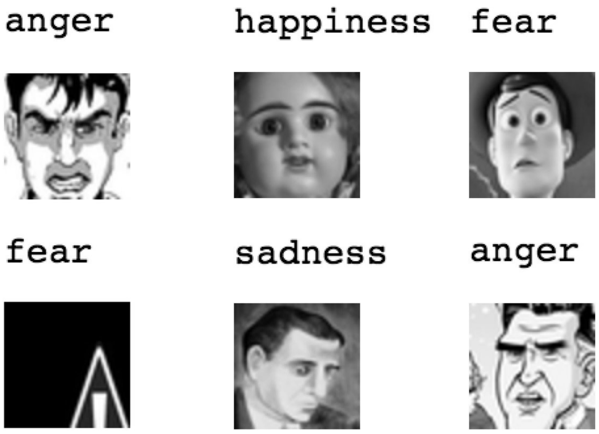


Figure 12: Images from the FER2013 data set showing some strange samples.

6 Results

In this section, the performance of different implementations of the program are compared. Implementations may vary because a different algorithm is used or because the same algorithm could be run with different configurations. For example, a convolutional neural network with a predetermined type of input may overall be treated as a single type of algorithm, however depending on how it is configured results are widely varied. The same convolutional neural network where each pixel of the image consists of part of the input can have different performances when the amount of layers is changed or the thresholds for activation fluctuate. When the input to the neural network changes it is already considered a change in the algorithm itself, not necessarily just configuration.

When looking at results it is important to first define which metrics are going to be used to determine the proficiency of the program. The main leverage used to arrive at these conclusions is how well the algorithm classifies the images left aside as the testing set. As these images are already labeled with the corresponding expression they can simply be sent as input to the program and compare the outputted expression with the actual one. The percentage of correctly classified images will be the main metric to take into consideration when determining how good the algorithm performs.

In this implementation no emphasis is made on how long the program takes to train itself nor how long it takes to classify an image. The main focus is on arriving at the correct classification of the image and not on how efficient it is while getting there. This could have been a valid point to take into account if the objective is to have a program that could process images into expressions in real time such as analyzing videos or just process significant amount of data as fast as possible. While the developed program is able of processing real time videos, in this results section no comparison will be made between algorithms nor configurations on how *efficient* they are, it will be purely on how *effective* they are.

One of the main ways that will be used to represent the results of image classification will be through confusion matrices. This consists of a square matrix where rows and columns are both the list of expressions which can be part of the output. For each classified image a counter will be increased in the cell matching the row of the actual expression the image represents (the labeled expression) and the column of the expression the program outputs.

	Happy	Sad	Surprised
Happy	9	0	1
Sad	0	10	0
Surprised	2	0	8

Table 1: Example of a confusion matrix where the testing data set consisted of 10 happy, 10 sad and 10 surprised images.

Taking as an example table 1, out of the ten images labeled as happy, nine are correctly classified and one is mistakenly classified as surprised. All ten sad images are correctly classified and two of the surprised images are classified as happy. The diagonal of the matrix is the total amount of correctly classified images while the total sum of the cells in the matrix is the total amount of images. Thereby the percentage of correctly classified images can be calculated using equation 2, where a_{ij} is the element in row i and column j of the confusion matrix.

$$\%correct = \frac{\sum_{i=0}^{n} a_{ii}}{\sum_{i=0}^{n}\sum_{j=0}^{n} a_{ij}} \tag{2}$$

6.1 Methods

In the following section all the implemented methods are discussed in regards to their results and steps taken to improve their performance. In each case the parameters used in each method are shown, be it a neural network structure or the different parameters HOG requires.

In all cases the data is split into two groups for testing purposes, one for training all the methods, and another one for testing whether the method is able to generalize the solution or if it simply memorized the solution for the training set.

Across all of the tests the training and testing subsets remain the same to be consistent with the results across algorithms. For training, the FER2013 data set is used because of its bigger size and more variety in the subjects and positions. For testing both the KDEF and FACES data sets are used. Both of these data sets have the subject right in front of the camera and making a specific expression upon request, so its better to determine if the expression can successfully be classified.

Then, the output of applying the specific method to the testing data set is shown, and some conclusions are given about the result of that test.

6.1.1 HOG + FFNN

One of the approaches is combining HOG as a feature detector and a FFNN as a classifier.

The HOG algorithm has several parameters that can be tweaked. One of these is the cell size which is the amount of pixels to be grouped for each bucket in the histogram. An example of applying HOG with different cell sizes can be seen on Figure 13.

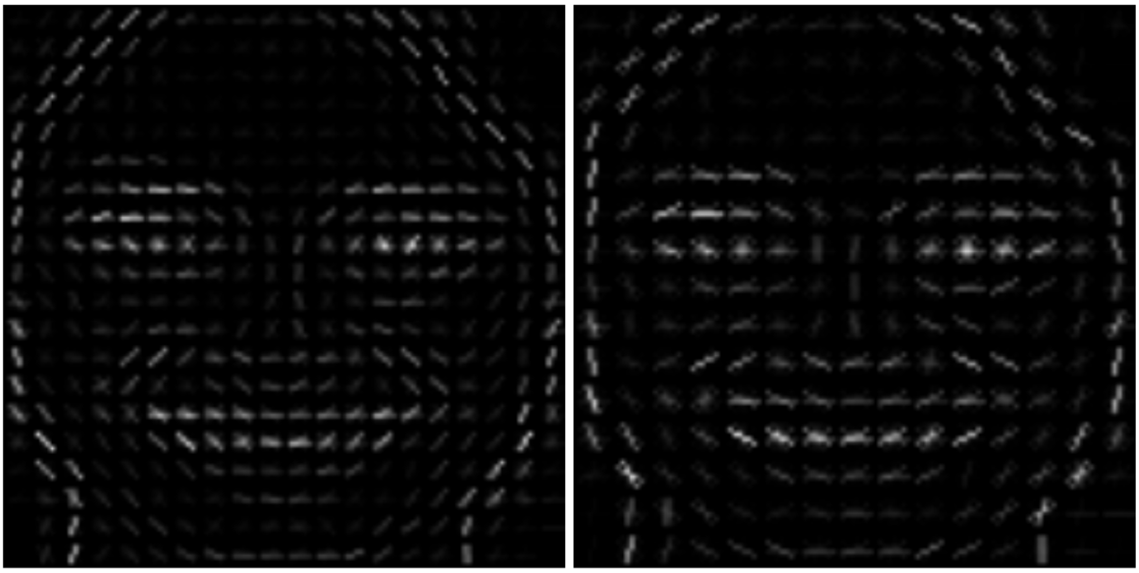


Figure 13: Superimposed image of applying HOG descriptor with cells of 6x6 and 8x8 respectively

It can be seen that with a smaller bucket size the details of detected borders is much higher. This is because as less pixels are grouped at a time then there is less loss of information. With the higher bucket size the detected borders are broader and further apart.

It cannot be said outright if one is better than the other. A smaller cell size means more buckets and hence more detail on the processed image which may outline some subtle features key to detecting accurately the expression. However the whole purpose of feature detection is extracting only the parts that would be considered important for classifying the expression. Therefore a case may be made as well for bigger cell size.

The following are the parameters used for HOG as listed in 4.1.2:

- window_size = (48, 48)
- cell_size = (8, 8)
- block_size = (24, 24)
- block_step = (8, 8)
- orientations = 9
- signed_gradients = True

To start off the FFNN is trained with only three expressions, happiness, sadness and surprise to see whether this method has any chance of success and is worth exploring further.

Because the size of the data generated by the HOG algorithm for this configuration is of length 1296 it is a concern that the network would over-fit. Over-fitting occurs is when the classifier is trained to specifically be able to classify the training data set and is not able to abstract a generalized solution for any other input. In other words, it memorizes the result for all the training data set without gaining any insight of what actually constitutes an expression. The starting configuration for the neural network can be seen on Figure 14. This does not avoid over-fitting as seen on the training accuracy on Figure 15 where the training accuracy reaches almost 100% but the testing accuracy starts to drop.

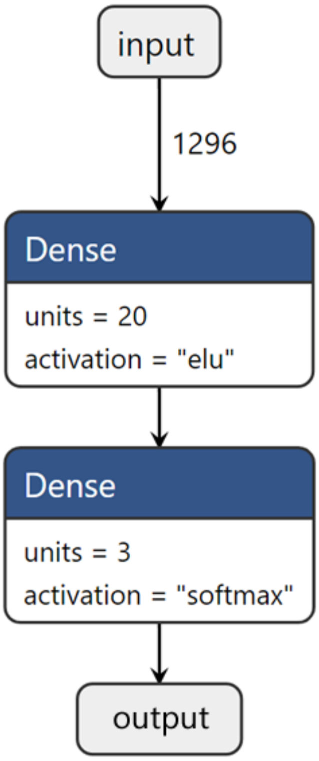


Figure 14: FFNN model for classification

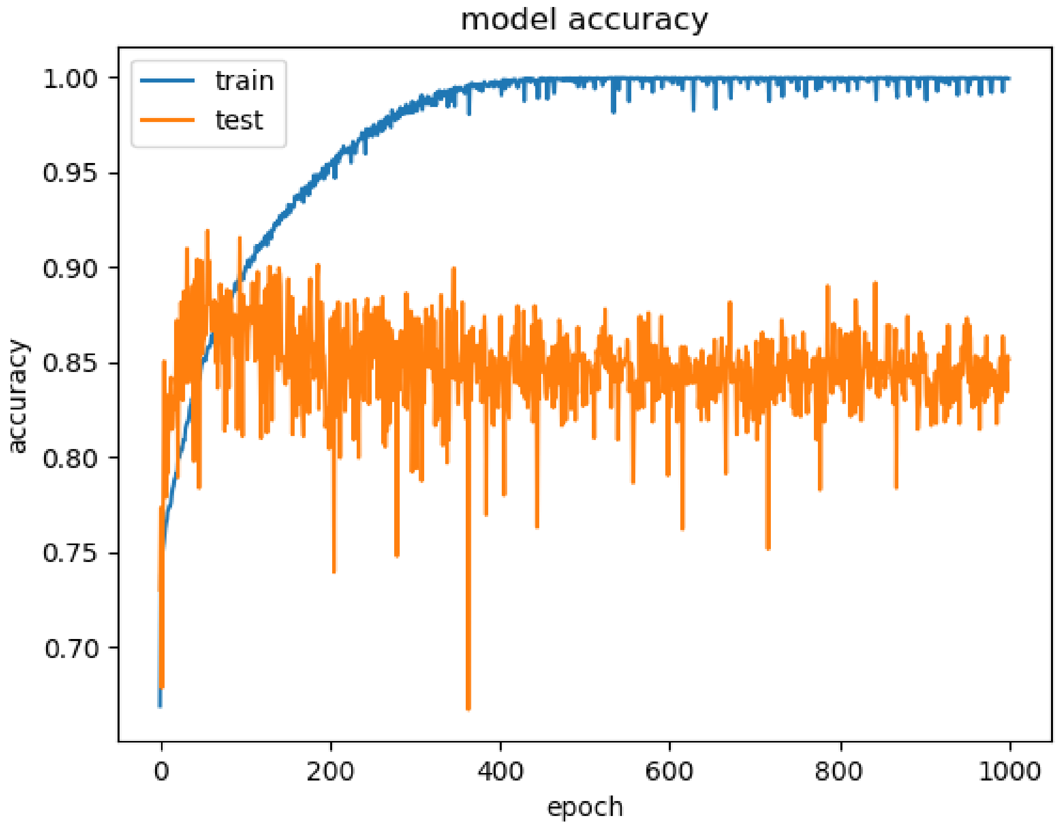


Figure 15: Training accuracy for the first model, which has over-fitting

To avoid this a new type of layer is added, called Dropout Layer. This layer helps with over-fitting by randomly dropping certain amount of neurons from the layer at the training phase, avoiding having the neurons memorize what to do for each of the training input and making them adapt to a partial loss in the information. With this new layer the configuration is as on Figure 16. The training accuracy can be seen on Figure 17 and as the figure shows, the training accuracy is below the testing accuracy, which implies the network is not over-fitting.

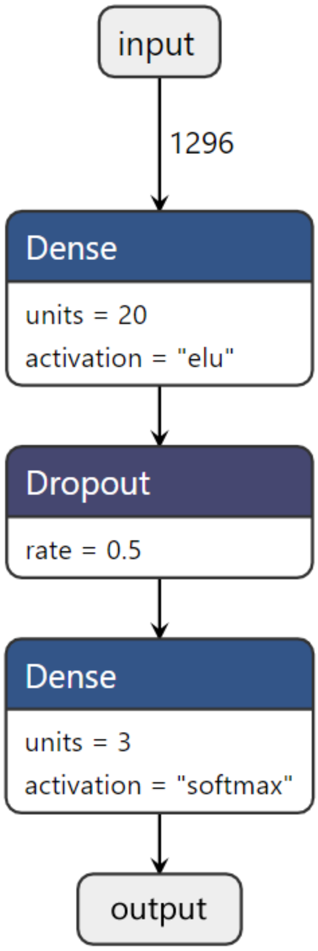


Figure 16: FFNN model for classification with a dropout layer added

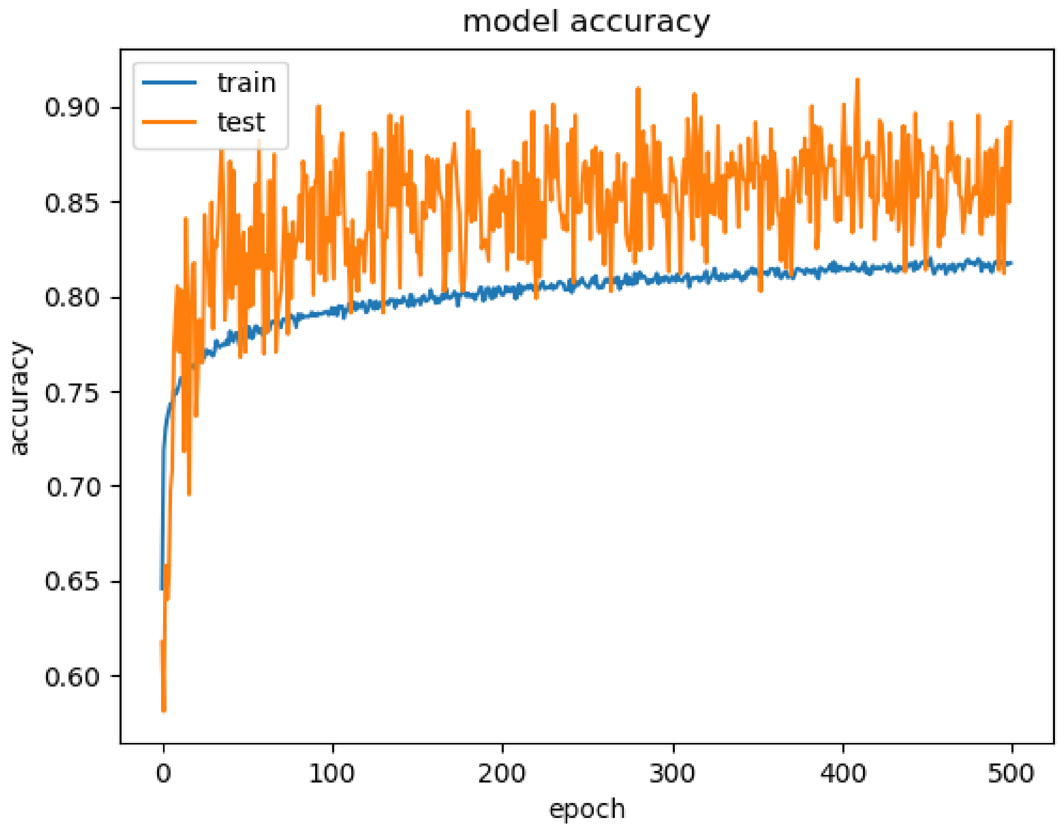


Figure 17: Training accuracy for the model with a dropout layer

The dropout rate used is 0.5. Table 2 shows the confusion matrix of results for HOG + FFNN configured as previously described.

Expression	Happiness	Sadness	Surprise
Happiness	455	0	0
Sadness	44	425	0
Surprise	15	32	93

Table 2: Confusion matrix for HOG with cell size of 8x8 and FFNN as classifier.

The proportion of correctly classified images is $\frac{973}{1064} = 0.9145$. The program correctly classifies all images depicting happiness but mistakes some that are supposed to depict sadness as happy. It struggles the most classifying the surprised images, having the lowest success rate for them at 66.43%.

Next the same configuration is tried but the HOG parameters are modified to use a smaller window size.

- window_size = (48, 48)
- cell_size = (6, 6)
- block_size = (12, 12)
- block_step = (6, 6)
- orientations = 9

- signed_gradients = True

The layout of the FFNN is left the same as Figure 16 with a dropout of 0.5 but now the input of the first layer is of size 1764, this is because having a smaller block step generates more buckets which increments the length of the input.

The confusion matrix for the updated configuration is shown on Table 3.

Expression	Happiness	Sadness	Surprise
Happiness	455	0	0
Sadness	52	417	0
Surprise	21	37	82

Table 3: Confusion matrix for HOG with cell size of 6 × 6 and FFNN as classifier

The proportion of correctly classified images now is $\frac{954}{1064} = 0.8966$. The same patterns as with a window size of 8 × 8 arise, with all happy images being classified as happy, some sad images being classified as happy as well and many surprised images being classified as the other two expressions.

Given that better results are achieved using HOG with a cell size of 8 × 8 than with 6 × 6 the next step is to try with an even bigger cell size. The following parameters for HOG are now used:

- window_size = (48, 48)
- cell_size = (12, 12)
- block_size = (24, 24)
- block_step = (12, 12)
- orientations = 9
- signed_gradients = True

This configuration results in an input length of 324, much smaller than the other configurations. Keeping the same layout for the network the following results shown on Table 4 are achieved.

Expression	Happiness	Sadness	Surprise
Happiness	455	0	0
Sadness	88	381	0
Surprise	37	28	75

Table 4: Confusion matrix for HOG with cell size of 12 × 12 and FFNN as classifier

The accuracy of the new model is only $\frac{911}{1064} = 0.8562$, which is the worst of all the configuration tested, but because the input is much smaller it is a valid consideration that the performance can be improved by making the network bigger. This is because compared with the previous attempts the input parameters are reduced the amount of information the network can store is greatly reduced. The configuration for the new neural network is shown on Figure 18. With this new layout the results for a cell size of 12 × 12 can be seen on Table 5.

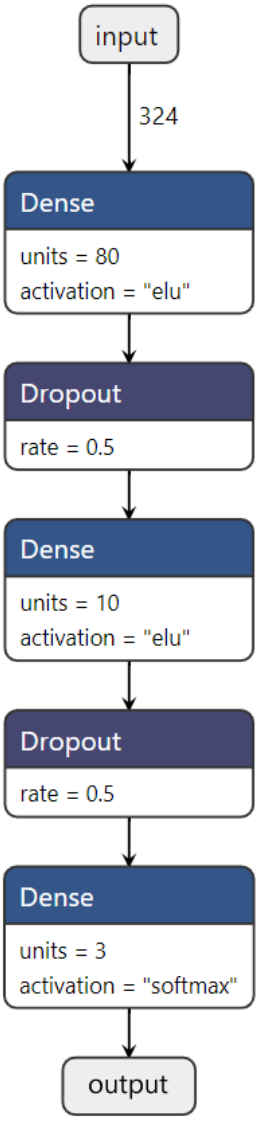


Figure 18: Bigger NN model to compensate smaller input

Expression	Happiness	Sadness	Surprise
Happiness	455	0	0
Sadness	67	402	0
Surprise	27	31	82

Table 5: Confusion matrix for HOG with cell size of 12x12 and FFNN as classifier.

The proportion of correctly classified images is now $\frac{939}{1064} = 0.8825$. Even though the performance for HOG with a cell size of 12×12 is improved with new configuration it still performs worse than with a cell size of 8×8. Once again the same patterns of misclassification that are previously pointed out for other cell sizes can be noticed.

Table 6 shows a summary of the performance of using HOG as feature detector with varying cell sizes and FFNN as a classifier. While all cell sizes arrive at roughly a similar proportion of correctly classified images (90%), the best performing cell size is that of 8×8.

Cell size	% of correctly classified
6×6	89.66
8×8	91.45
12×12	88.25

Table 6: Performance of program using HOG and FFNN with different cell sizes

6.1.2 One FFNN per expression

Another approach is to train a single FFNN for each expression. The parameters used for all feed forward neural networks are the ones in Figure 19. In this case it is not a single neural network being trained to classify the expression in an image but instead one neural network is trained per expression.

Each network outputs the probability that the expression is present in an image. The expression with the highest probability is then the one which the images gets classified as.

Another option is, instead of only returning the expression with the best probability, return all the expressions with a probability above certain threshold, which allows a more complex classification.

In Tables 7, 8 and 9 the confusion matrix for each of the emotions can be seen. Their accuracies are $\frac{1019}{1064} = 0.9577$, $\frac{991}{1064} = 0.9313$ and $\frac{1011}{1064} = 0.9501$ for happiness, sadness and surprise respectively. This accuracy is much better than the one of the single neural network. When put together and picking the expression with the best probability the following confusion matrix as seen on Table 10 is achieved, with an accuracy of $\frac{980}{1064} = 0.9211$. This is worse than any of the individual networks, but still better than using a single neural network.

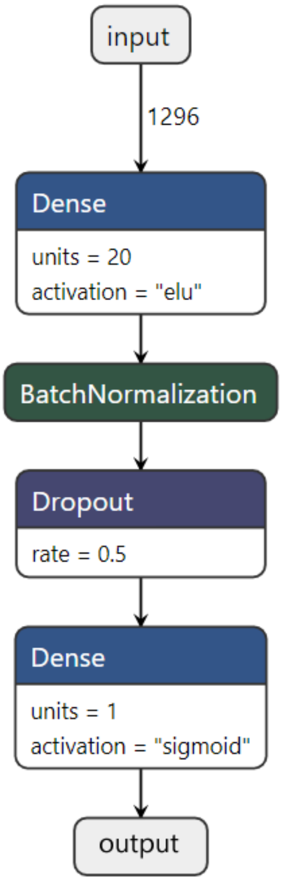


Figure 19: Model used to train each of the emotions by themselves

Expression	Happy	Not Happy
Happy	455	0
Not Happy	45	564

Table 7: Confusion matrix for Happy

Expression	Sad	Not Sad
Sad	433	36
Not Sad	37	558

Table 8: Confusion matrix for Sad

Expression	Surprised	Not Surprised
Surprised	87	53
Not Surprised	0	924

Table 9: Confusion matrix for Surprised

Expression	Happiness	Sadness	Surprise
Happiness	455	0	0
Sadness	31	438	0
Surprise	18	35	87

Table 10: Confusion for picking the emotion with the best probability

After obtaining more than 90% accuracy for the 3 expression, the next step is to attempt to add another one, in this case anger is chosen. To add the new expression it is necessary to re-train the previously created networks with a new set of images containing angry people. Using the same layout for the neural network and the same parameters for HOG the following confusion matrix for each of the expressions are achieved as seen in Tables 11, 12, 13 and 14.

Expression	Happy	Not Happy
Happy	454	1
Not Happy	80	1000

Table 11: Confusion matrix for Happy

Expression	Sad	Not Sad
Sad	271	198
Not Sad	180	886

Table 12: Confusion matrix for Sad

Expression	Surprised	Not Surprised
Surprised	87	53
Not Surprised	0	1395

Table 13: Confusion matrix for Surprised

Expression	Angry	Not Angry
Angry	242	229
Not Angry	81	983

Table 14: Confusion matrix for Angry

Expression	Happy	Sad	Surprised	Angry
Happy	455	0	0	0
Sad	56	333	1	79
Surprised	18	8	98	16
Angry	34	178	0	259

Table 15: Confusion matrix for all 4 expressions

The confusion matrix when testing with all expressions is displayed in Table 15 .

It can be seen the networks have a hard time distinguishing between sad and angry, obtaining 75% and 79% accuracy respectively. Even though Happy and surprised maintained their accuracy when adding a new expression, the errors between sad and angry are enough to decrease the total accuracy from over 90% to only 74.6%. In order to figure out why this could be happening the data sets are thoroughly examined. One possible reason is that some images such as shown in Figure 20 are labeled as anger, but to the human eye they could be confused with sadness.

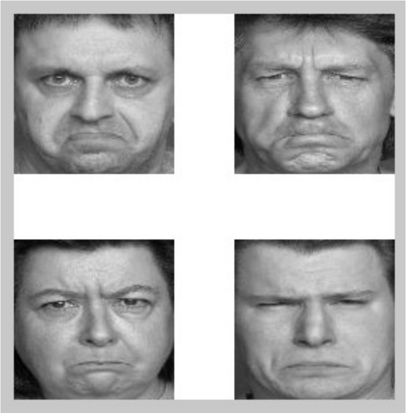


Figure 20: Images whose faces are labeled as anger but may be confused with sadness

Because of this significant decrease in accuracy the approach is reevaluated to stop trying to add new emotions and instead try different methods in hope of improving the performance.

6.2 Gabor + FFNN

The next configuration tested is using Gabor as a feature detection method and keep using a FFNN as a classifier.

First Gabor is computed for each of the directions, then a new image is generated by the superposition of all the generated images as shown on Figure 4. The result is then flattened into a one dimensional array with the values normalized between 0 and 1.

The model for the neural network can be seen of Figure 21.

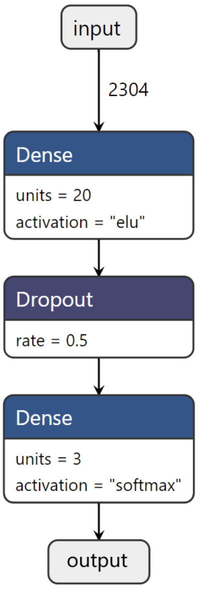


Figure 21: FFNN model for Gabor

This is the same configuration as the one used for HOG, with the only difference being the size of the input, which is now 48 × 48 = 2304. The result of the classification can be seen on Table 16.

Expression	Happiness	Sadness	Surprise
Happiness	415	40	0
Sadness	83	385	1
Surprise	17	88	35

Table 16: Confusion matrix for Gabor + FFNN

With an accuracy of only 78.47% Gabor performs worse than HOG while using the same classifier. Therefore it is decided not to explore Gabor as a feature detector further.

6.3 Convolutional Neural Network

Convolutional Neural Networks (CNN) are a straightforward choice when trying to classify images. This is because of their ability to take images as input and maintain a spacial relation across the classification. On the contrary when using FFNN with an image as an input all the information about the position of that pixel and the values of the pixels near them are lost. Because CNN works by performing a convolution and applying a filter to every pixel, the filter takes into account the surrounding pixels. This is the main reason why convolutional neural networks are the go-to method of image classification. The CNN layout is on Figure 22. How each of the layers works can be read on Section 4.2.2

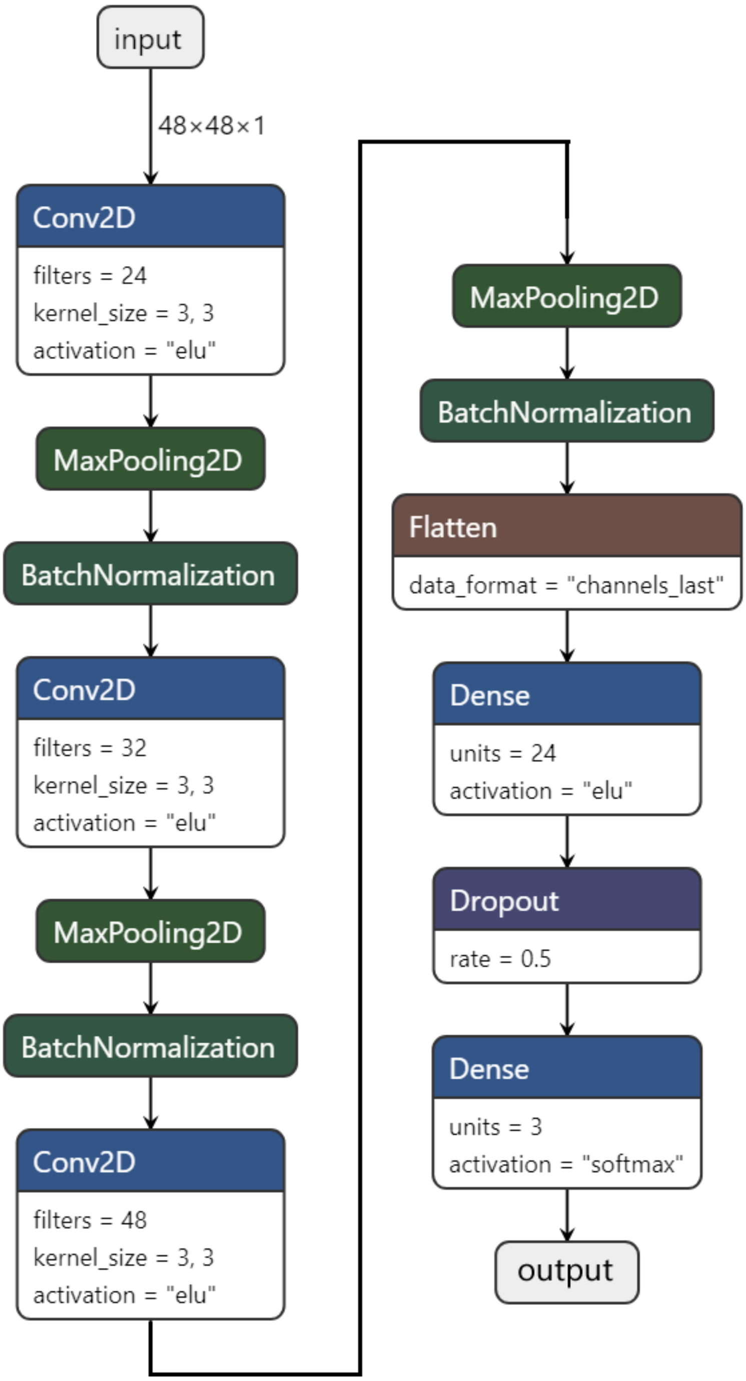


Figure 22: CNN layout

The result of the training shows an accuracy of 91.63% and the confusion matrix can be seen on Table 17.

Expression	Happiness	Sadness	Surprise
Happiness	450	5	0
Sadness	23	446	0
Surprise	20	41	79

Table 17: Confusion matrix for the CNN

Even though the accuracy is on par with the previous method, it can be seen in Figure 23 that the CNN over-fits in a short amount of epochs. Batch Normalization in the convolutional layers and Dropout in the dense layers are not enough to avoid the over-fitting issues, so another method is implemented called Data Augmentation. The follwing are the parameters for data augmentation as listed on Section 4.2.2:

- rescale $= \frac{1}{255}$ so that the values from the images goes from 0 to 1, instead of 0 to 255
- rotation range = 45 degrees
- shear range = 0.2
- horizontal flip

With this data augmentation the network does not over-fit during training as seen on Figure 24 and the accuracy of the model reaches 94.64% with the confusion matrix shown on Table 18.

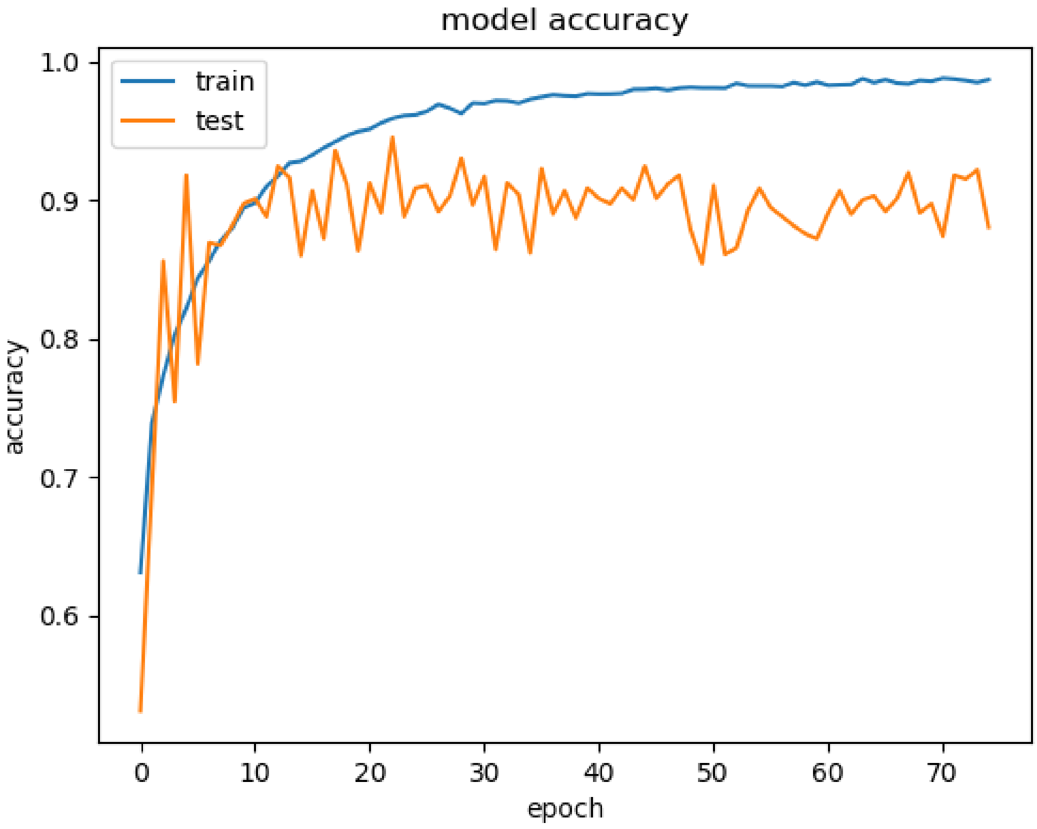


Figure 23: Training accuracy for the CNN

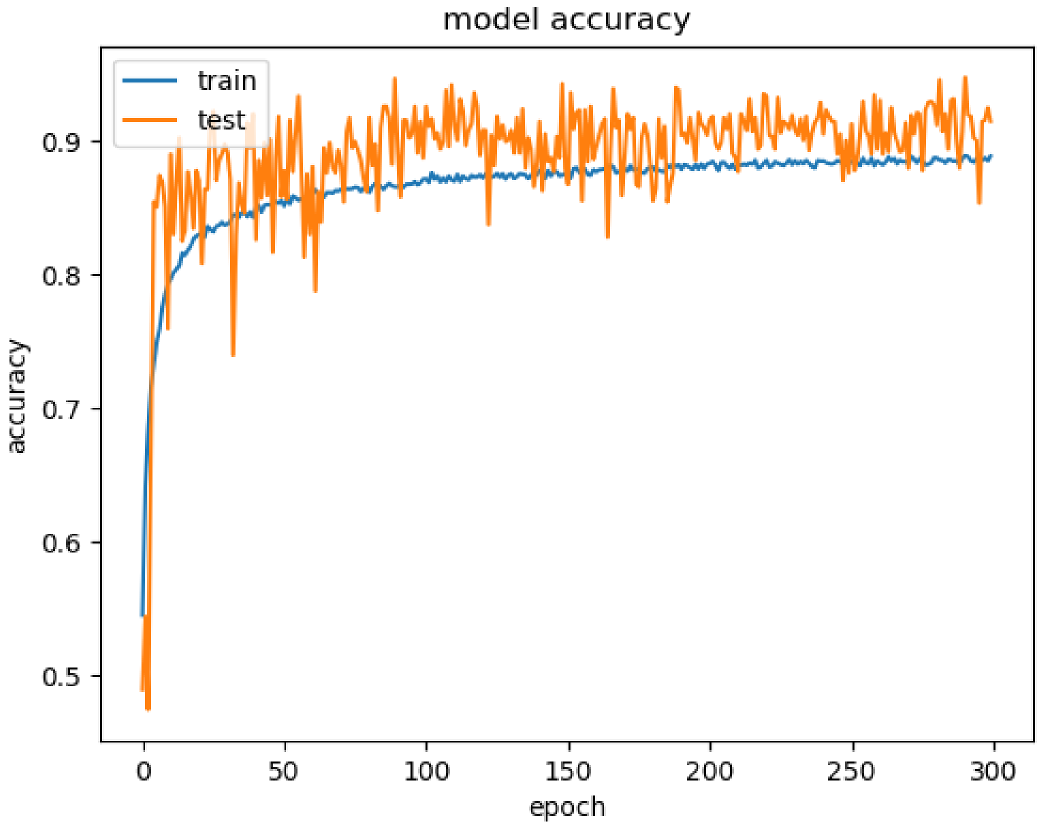


Figure 24: Training accuracy for the CNN with data augmentation

Expression	Happiness	Sadness	Surprise
Happiness	453	2	0
Sadness	31	437	0
Surprise	7	16	117

Table 18: Confusion for the CNN when using data augmentation

While convolutional have performed better than the previous methods discussed, further attempts to improve them can be tried such as doing the same thing as with feed forward neural networks and have one CNN per expression. Keeping the same layout for each of the expression as the one on Figure 25 the confusion matrices shown on Tables 19, 20 and 21 are achieved.

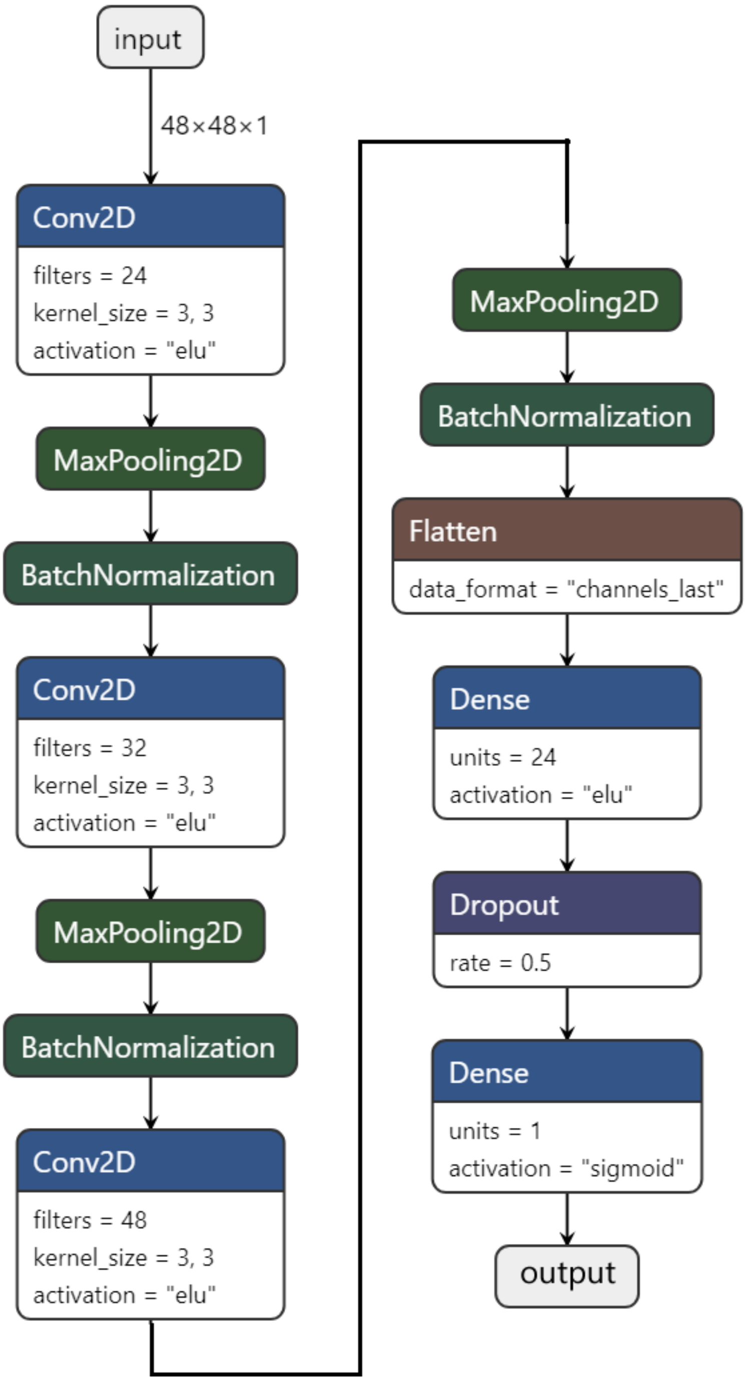


Figure 25: CNN layout for each of the expressions

Expression	Happy	Not Happy
Happy	448	7
Not Happy	3	606

Table 19: Confusion matrix for Happy CNN

Expression	Sad	Not Sad
Sad	450	19
Not Sad	30	565

Table 20: Confusion matrix for Sad CNN

Expression	Surprised	Not Surprised
Surprised	119	21
Not Surprised	8	916

Table 21: Confusion matrix for Surprised CNN

The CNN trained to detect happiness has a success rate of $\frac{1054}{1064} = 0.9906$, for sadness $\frac{1015}{1064} = 0.9539$ and for surprise $\frac{1035}{1064} = 0.9727$.

When combining all the CNNs and returning the expression with the highest probability an accuracy of 97% is reached, with the confusion matrix shown on Table 22

Expression	Happiness	Sadness	Surprise
Happiness	453	2	0
Sadness	12	451	0
Surprise	2	9	129

Table 22: Confusion for all the CNNs together

After the success convolutional neural networks show over feed forward neural networks the next step is to try once again to include anger and see if this method proves to be able to better differentiate sadness from anger. To achieve this, the sadness CNN is also trained with anger images in the data set and the anger CNN is trained with all the other expressions. The accuracy from the CNN with the task of determine whether a expression is sad or not goes down from the 95% success rate prior to 86% with the new confusion matrix shown on Table 23 and the accuracy of the Angry CNN is 88% while its confusion matrix is shown on Table 24.

Expression	Sad	Not Sad
Sad	341	128
Not Sad	85	981

Table 23: Confusion matrix for Sad CNN, now with angry in the data set

Expression	Angry	Not Angry
Angry	340	131
Not Angry	53	1011

Table 24: Confusion matrix for Angry CNN

Attempting to classify all the four expressions in the testing data set ends up with an accuracy of 87.62% with a confusion matrix than can be seen on Table 25.

Expression	Happy	Sad	Surprised	Angry
Happy	453	2	0	0
Sad	10	395	5	59
Surprised	2	9	128	1
Angry	6	91	5	369

Table 25: Confusion matrix for all 4 expressions

7 Conclusions and future work

Across all tested methods the performance is similar for identifying particular expressions. Happiness is always the most successfully classified expression and all methods have the worst performance when it comes to differentiating surprise from other expressions. When anger is included in the set of expressions it is often confused with sadness and vice versa, lowering the overall performance of the method.

The data sets play an important role in the whole expression detection process as they are the source from which the classifiers learn to differentiate expressions and they are used to test their effectiveness. Two of the used data sets, KDEF and FACES, have a normalized way of taking their pictures, with a standard distance to the camera, same subjects for all expressions and same image quality. However they are relatively small data sets, with 980 and 2052 frontal facing images respectively. The FER2013 data set is comparatively much larger with 25709 images, however by its nature of coming from a combination of different sources from the internet the quality of it is lower. With a high percentage of the images being of lower quality the results can be skewed by bad samples and be another factor why some expressions had much difficulty being told apart from one another.

Happiness is usually expressed by humans in the form of an upwards pulling of the edges of the lips which frequently exposes the teeth. This trait is not shared with any other tested expression and is what makes happiness the best performing expression of all. Other expressions share some characteristics which leads to confusion between them. Anger and sadness share the characteristic of frowning, lips pulled down and closed mouth.

In regards to neural networks, the best approach for convolutional and feed forward neural networks is to train a single neural network per expression. Then the image is classified by running it through all networks and choosing the expression which outputs the highest probability. The most challenging part of training the neural networks is to avoid over-fitting, and combination of dropout, normalization and augmentation help to avoid this problem. It is expected for convolutional neural networks to perform better than feed forward neural networks. This is because convolutional neural networks keeps information about the location of the pixel with respect to its neighbours while feed forward networks don't. Results show that convolutional neural networks indeed perform better and are proven to be better suited for expression detection through image analysis.

One point to improve on is to get a more robust data set, such as maintaining the quality of the KDEF and FACES data sets but with more subjects. If the overall performance of methods can be improved with this new data set then an attempt can be made to add more expressions to the program.

Further work can also be made into the use of other type of classifiers such as unsupervised learning ones and the use of different type of feature detectors.

References

[1] Faris A. Abuhashish, Jamal Zraqou, Wesam Alkhodour, Mohd Shahrizal Sunar, and Hoshang Kolivand. Emotion interaction with virtual reality using hybrid emotion classification technique toward brain signals. In *International Journal of Computer Science and Information Technology (IJCSIT)*, volume 7, pages 159–182, 2015.

[2] Hsu Chih-wei, Chang Chih-chung, and Lin Chih-jen. A practical guide to support vector classification, 2010.

[3] Chin-Teng Lin and Chao-Hui Huang. A complex texture classification algorithm based on gabor-type filtering cellular neural networks and self-organized fuzzy inference neural networks. In *2005 IEEE International Symposium on Circuits and Systems*, pages 3942–3945 Vol. 4, May 2005.

[4] Jie Deng. Emotion-based music retrieval and recommendation. *Open Access Theses and Dissertations*, 2014.

[5] B. Fasel and Juergen Luettin. Automatic facial expression analysis: a survey. *Pattern Recognition*, 36(1):259 – 275, 2003.

[6] The Royal Society For the Prevention of Accidents. Driver fatigue and road accidents, 2017.

[7] Mohammad Haghighat and Ehsan Namjoo. Evaluating the informativity of features in dimensionality reduction methods. In *Application of Information and Communication Technologies (AICT) 2011 5th International Conference on*, 10 2011.

[8] Mohammad Haghighat, Saman Zonouz, and Mohamed Abdel-Mottaleb. Identification using encrypted biometrics. In Richard Wilson, Edwin Hancock, Adrian Bors, and William Smith, editors, *Computer Analysis of Images and Patterns*, pages 440–448, 2013.

[9] Samin Iftikhar, Rabia Younas, Noshaba Nasir, and Kashif Zafar. Detection and classification of facial expressions using artificial neural network. *International Journal of Information Technology and Electrical Engineering*, 3:18–22, 2014.

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[11] Qiang Ji, Zhiwei Zhu, and P. Lan. Real-time nonintrusive monitoring and prediction of driver fatigue. *IEEE Transactions on Vehicular Technology*, 53(4):1052–1068, 2004.

[12] Z. Kasiran and S. Yahya. Facial expression as an implicit customers' feedback and the challenges. In *Computer Graphics, Imaging and Visualisation (CGIV 2007)*, pages 377–381, 2007.

[13] Jyoti Kumari, R. Rajesh, and K.M. Pooja. Facial expression recognition: A survey. *Procedia Computer Science*, 58:486 – 491, 2015.

[14] Orion Market Research Pvt. Ltd. Global virtual reality gaming market research and forecast 2018-2023, 2018.

[15] Flykt A. Lundqvist, D. and A. Öhman. The Karolinska Directed Emotional Faces - KDEF, 1998.

[16] Michael J. Lyons, Julien Budynek, and Shigeru Akamatsu. Automatic classification of single facial images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(12):1357–1362, December 1999.

[17] L. Ma and K. Khorasani. Facial expression recognition using constructive feedforward neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(3):1588–1595, 2004.

[18] S. Marĉelja. Mathematical description of the responses of simple cortical cells∗. *J. Opt. Soc. Am.*, 70(11):1297–1300, Nov 1980.

[19] Albert Mehrabian. *Nonverbal Communication*. Taylor and Francis, 2007.

[20] Dr. Parag Kulkarni Mrs. Ayesha Butalia, Dr. Maya Ingle. Facial expression recognition for security. *International Journal of Modern Engineering Research (IJMER)*, 2:1449–1453, 2012.

[21] Michaela Riediger Natalie C. Ebner and Ulman Lindenberger. FACES, 2005-2007.

[22] International Federation of the Phonographic Industry IFPI. Global music report 2018, 2018.

[23] Jun Ou. Classification algorithms research on facial expression recognition. *Physics Procedia*, 25:1241 – 1244, 2012.

[24] Ben Popper. This is how Facebook will animate you in VR, 2016.

[25] Mohammed Owais Qureshi and Rumaiya Sajjad Syed. The impact of robotics on employment and motivation of employees in the service sector, with special reference to health care. *Safety and Health at Work*, 5(4):198 – 202, 2014.