RECOGNITION OF FACIAL EXPRESSIONS FOR MAPPING EMOTIONS

Beatriz Carvalho[1] , **Inês Bem-Haja**[1] , **Mário Jorge Simões**[1]
[1]Department of Physics, Faculty of Sciences and Technology, University of Coimbra, 3004-516, Coimbra, Portugal.

Abstract

Nowadays, facial recognition systems have become more and more a part of society, with many applications to its name. In this project, the link between facial and emotional recognition was explored. Emotional recognition is an important social competence that empowers the individual to respond adequately to the environment. It is through facial expressions that this link is able to exist, since they are representative of people's mental and psychological state. This may have consequences in their daily actions. Therefore, some emotions were identified, namely anger, disgust, fear, happiness, sadness, surprise and neutral, through the analysis of a dataset of photographs that contain facial expressions representing these same emotions.

1 Introduction

Facial detection and recognition has been around since the 1960s yet very crudely. It was not until the 1980s that a massive leap in this matter was made. This technique can be used in a multitude of areas such as retail, access and recognition and even aviation and transport [1]. However, what if we could use this technology to aid in the recognition, diagnose and treatment of mental illnesses.

Over the years, mental health has gone from a stigmatised subject to a movement of bettering oneself within the younger generation. Mental health determines how the world is perceived, making it the foundation of emotions and respective responses [2]. Around the world, over 15% of the population is affected by mental illness and that statistic is bound to continue increasing. For example, the World Health Organisation (WHO) estimates that unipolar depressive disorders will be "the leading cause of the global burden of disease" by 2030 [3].

Unfortunately, regarding the integration of new diagnostic techniques in psychiatry, there is still a lot of progress to be made [4]. When talking about physical illnesses there is a number of possible ways to not only help conduct but also diagnose a patient. However, and possibly as a result of years of stigma, the diagnosis of mental illnesses rests solely on the shoulders of the patients' own expression and psychiatrists' evaluations. This results in human error that can amount to years in therapy and/or incorrect diagnosis due to a plethora of reasons such as the overlapping symptoms in different mental illnesses.

It is here where the ever-growing field of facial recognition can help bridge this gap. Usually, when talking about artificial intelligence (AI) in psychiatry, most algorithms are designed to evaluate the DSM-5 criteria [4]. The Diagnostic and Statistical Manual of Mental Disorders (5th ed.), also known as DSM-5, is a reference guide for mental health clinicians to diagnose, classify, and identify mental health conditions [5, 6].

Combining both algorithms of evaluating, the DSM-5 criteria and facial/emotion recognition would be the ideal situation with the most reliable results. However, there is still a lot of research to be done when talking about facial/emotion recognition. Therefore, our paper will only look at how to improve this algorithm.

The route we took consisted on training a dataset that contained images with seven different types of emotions: anger, disgust, fear, happy, sad, surprise and neutral. To do so we took advantage of Convolutional Neural Networks (*CNNs*) which has shown great potential due to their powerful automatic feature extraction and computational efficiency. In our work, we adopt the *CNN* architecture, fine-tune rigorously its hyperparameters, analysing and identifying all the present emotions whilst optimising training using *Keras*.

Our dataset contained a multitude of images, nearly 36000 files, which means that the complexity of the training procedures increases exponentially when compared to previous works [7]. Nevertheless, by using this set of tools, we were able to expedite the training speed of the algorithm. The overview of the current system is demonstrated in Figure 1.

The remainder of this paper is structured as follows: in Section 2 we discuss published work that relates to our project; in Section 3 we describe the data we are working with; in Section 4 we detail how the research was carried out; in Section 5, the results of the different models for the 7 emotions are presented. Finally, in Section 6, some conclusions are drawn.

2 Related Work

Convolutional Neural Networks or *CNNs* were introduced in the late 1990s but they were limited due to the lack of training data and computational power. After 2010s with the growth

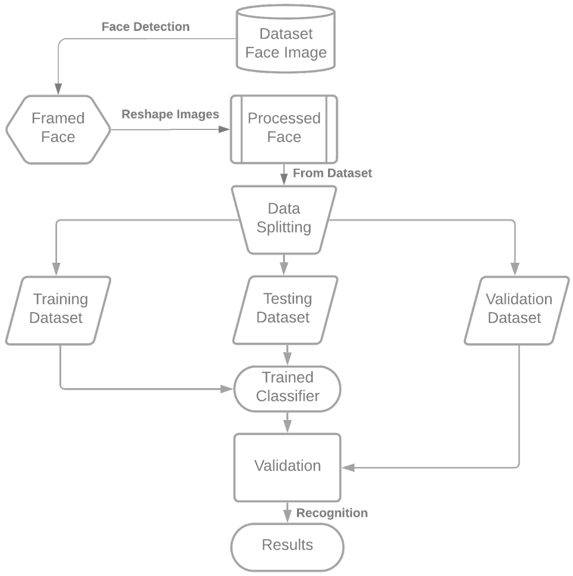

Figure 1: System's overview.

of computing power and the collection of larger datasets, they have been able to achieve results which were previously considered to be purely within the human realm [8].

Regardless, there are still some issues that involve this model, such as overfitting. For context, overfitting occurs when a model trains the data too well. It happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data [9]. Because of this, various techniques have been proposed to further improve performance. For instance, when it comes to overfitting, the dropout, regularisation, and data augmentation are used to prevent it. Also, by using an Adam Optimiser instead of a GSD (Stochastic Gradient Decent), it allows for a faster training since it uses techniques like momentum and adaptive learning rates [10]. Batch normalisation has also been developed to help prevent gradient vanishing and exploding [11].

With the developments and extensive research on *CNNs* they are now seen as a much more viable tool in feature extraction, pattern recognition and image processing. Many *CNN* variants have achieved remarkable results with a classification accuracy between 65% to 95% [12, 13]. Different architectures performance have also been studied. Actually when comparing *VGG*, *Inception* and *ResNet* it was shown that *VGG* had an accuracy of 72.7%, followed by *ResNet* at 72.4% and *Inception* at 71.6% [14]. However, of all the papers viewed, it seems disgust is always the emotion with the lowest results, meaning it is unrecognised.

Moreover, it has been shown that performance is improved when multiple different models are ensembled [14, 15]. Nonetheless, we first aim to optimise a single network, which then allows one to improve the performance of the ensemble even further.

3 Materials

During the early stages of our project, we encountered a challenge proposed on *Kaggle* - a platform that allows its users to host data science and machine learning competitions [16] - titled "Facial Keypoint Detection" by Octávio Souza. On the platform, they try to predict emotions through facial expressions using *Keras* and *VGG16*. Given that we had some difficulty running the code, we needed to change our approach.

Following a more detailed search, we found another challenge on *Kaggle* with the same purpose of predicting emotions through facial expressions, this time using *Convolutional Neural Network (CNN)* and *Keras*. It is titled "Challenges in Representation Learning: Facial Expression Recognition Challenge" by Pierre-Luc Carrier and Aaron Courville [17].

3.1 Frameworks

Before moving on to the following sections, it is essential to first explain what *CNN* and *Keras* are.

CNN is a Deep Learning algorithm which can take in an input image or the raw pixel data, assign importance to various aspects in the image (ergo training the model) and be able to differentiate one from the other automatically extracting the features for better classification. This way it allows the user to extract higher representations for the image content [18, 19].

On the other hand, *Keras* is a deep learning API written in *Python* that was developed with a focus on enabling fast experimentation [20].

Keras runs on top of the machine learning platform *TensorFlow*, one of the several *Python* libraries we used. More specifically, it is an open source machine learning library applicable to a wide variety of tasks. It is a system for creating and training neural networks to detect and decipher patterns and correlations, analogous to (but not equal to) the way humans learn and reason [21].

Additionally, we used the *NumPy* and *Pandas* libraries because they support the processing of large, multi-dimensional arrays and matrices, along with a large collection of high-level math functions to operate on these matrices [22, 23, 24]. Furthermore, *Seaborn* and *matplotlib* libraries were used in order to conduct data visualisation.

3.2 Jupyter Lab

Initially, to take advantage of all the libraries mentioned in Section 3.1, we started by using *Google Colab* as a code compilation tool since it allows users to write and execute arbitrary *Python* code through the browser [25]. It is extremely well suited for machine learning which is exactly what we are going to work with. As this tool caused some complications while trying to process a few *Keras* libraries, we decided to switch to the *Jupiter Lab* platform to run the algorithm. After all *Google Colab* is a hosted *Jupyter* notebook service that requires no setup to use, which meant that it resulted in a smooth transition from *Colab* to *Jupiter Lab*.

3.3 Data Description

As previously mentioned in Section 3, our work is based on another research project by Pierre-Luc Carrier and Aaron

Courville, which means that they were the ones that provided and prepared the dataset for us.

The data consists of 48x48 pixel images of faces presented in a grayscale spectrum. People's faces have been automatically recorded to be more or less centred and to occupy approximately the same space in each image. Our goal is to classify each face into seven categories, based on the emotions displayed in their respective facial expression: 0 = angry, 1 = disgust, 2 = fear, 3 = happy, 4 = sad, 5 = surprise and 6 = neutral [17].

In the task, there is a class named "train.csv" that contains two columns, "emotion" and "pixels". The "emotions" column contains a number code between 0 and 6, including the emotion present in the image, whereas the "pixels" column contains a string surrounded by quotes for each image. The content of this string is a space-separated pixel value in the main order of the row, and the task is to predict the emotion column [17].

It is important to note that the training set consists of 28,709 examples, the public test set consists of 3,589 examples, and the final test set consists of another 3,589 examples, which represents a sizeable amount of information.

4 Methods and Approach

Our methodology consists of organised steps, much like our code, so that it is easier to not only read but understand the process. It consisted in: importing packages, definition of functions, data preparation and model definition and training. Every step is carefully explained in the following sections.

4.1 Import of the Necessary Libraries and Packages

The first step was to import the necessary packages to *Jupiter Lab* in order to carry out our research. Beforehand, in Section 3.1, we have introduced some of them (*NumPy*, *Pandas*, *matplotlib.pyplot* and *Keras*) and explained them to a certain extent. We should, however, clarify the specific role they played in the building of this algorithm: *Pandas* is a Python-based data analysis toolkit [24], *NumPy* allows imports to be linked to the selected local variable name, usually to avoid name collisions [23], *matplotlib* allows a better visual representation of the plots and figures [26], and lastly *keras* allows the training framework to different imported layers and models [20].

In addition, we also imported *os* which provides functions for creating and removing a directory [27], and *sklearn.metrics* and *mlxtend.plotting* were both used for the development and representation of a confusion matrix [28, 29].

4.2 Definition of Useful Functions

Analysing our work, it is straightforward to identify our structure and organisation. The code is divided in functions with the purpose of facilitating the understanding of the algorithm.

It is not typical of a code to explain what each of its function is supposed to do. We tend to run the code and view the results, and so, with that in mind, some background information is necessary in order to serve as a further aid.

The first function is *prepare_data(data)* which is used to get an image array and its corresponding label. *plot_examples(label=i)* and *plot_all_emotions()* will plot five images with the emotion i $\in$ [0;6] label and seven images of the seven types of emotions considered, respectively.

Additionally, *plot_image_and_emotion(test_image_array, test_image_label, pred_test_labels, image_number)* plots the image and compares the prediction results with its respective label, and lastly *plot_compare_distributions(array1, array2, title1=", title2=")* is used to plot a figure in order to compare the different expression counts in the used dataset.

4.3 Dataset and Data Preparation

As mentioned in Section 3.3, our data contains images already labelled along with 35887 files, we had to now import, prepare and preprocess the dataset available in the *Kaggle* challenge.

The next step was to use the suggested data usage, splitting the training data in three parts: *Training*, *PrivateTest* and *PublicTest*.

4.4 CNN Model Definition and Training

Previously, in Section 3.1, we have introduced what *CNN* is and how it works. Even so, there are a few concepts that we reserved for this section. As we mentioned before, *CNN*, or Convolutional Neural Network, is a machine learning algorithm widely used in AI.

Firstly, this network uses *ConvNet* for image processing. This serves to reduce images to a certain size, so that they are easier to process, without the lost fundamental resources to achieve a good prediction [19].

This architecture features several layers: convolution layer (kernel), pooling layer, and FC (fully connected) layer. The first layer essentially serves for extracting the resources of the image. The second is responsible for reducing the spatial size of resources, which is important because it decreases the computational power required for data processing. The third and final layer is usually used to learn nonlinear combinations of resources [19].

We defined the main parameters of the CNN model as the convolutional layers and the *softmax* and *relu* activation function. After this, we are now in the conditions of training the model using *model.fit()* [30].

4.5 Determination of Number of Epoch

Whilst developing the algorithm, we did start training the model with a number of 20 epochs even so, as we will further explain in Section 5, the optimal epoch number for our project is 9.

For clarification, an epoch defines the number of times that the learning algorithm will work through the entire training dataset. One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters [31].

5 Results and Discussion

At this point, we can test the implementation of the proposed emotion recognition system based on facial recognition. Simultaneously, we can analyse the graphics obtained from the

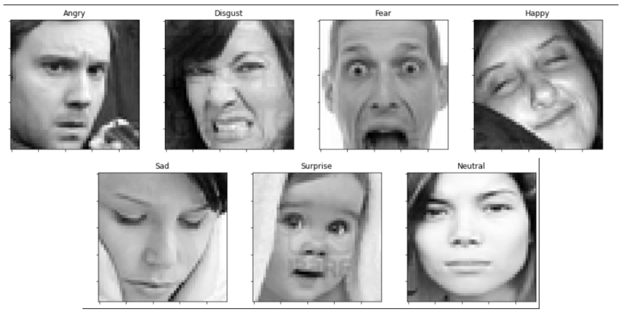

Figure 2: Examples of all the facial expressions considered.

model, which relate to the model's parameters, such as loss, validation loss, accuracy and validation accuracy, depending on the number of epochs trained.

In Figure 2 we can find an image for each of the 7 categories of emotions present in this dataset: angry, disgust, fear, happy, sad, surprise and neutral.

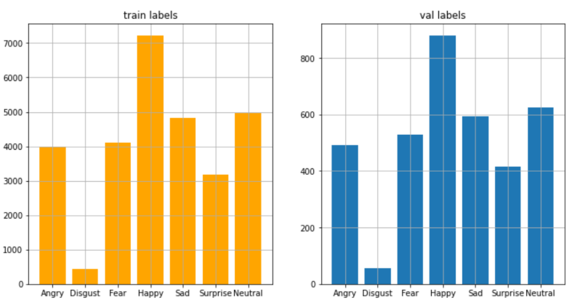

Figure 3: Comparison of the counts of the facial expressions labelled in the training model.

Figure 3 contains two bar charts for train labels and lab labels, respectively. The first chart is constructed based on the set of data reserved for training whilst the second is composed of validation data. Usually data is split into only 2 sets: training data and test data. However, this split may not be enough since the stability and performance of the model will have to be "tested" with the training data set. Thus, in order to try to avoid overfitting, the data is divided into the 3 sets. A percentage of the data is then stored for validation, providing an impartial assessment of a model fit. By looking at and comparing the two charts, we can conclude that the number of images in each category is virtually the same in the test dataset and validation dataset.

After calculating, in 20 epochs, the results of loss and accuracy in both validation and test, we were able to obtain the final value of the accuracy of the test, having achieved a percentage of 55,2%. Even though this is not a very high value in and of itself, the fact that it is higher than 50% may indicate that this model classifies images where emotions are expressed [32].

Figures 4 and 5 contain two charts with the values of accuracy and loss as functions of the number of epochs, therefore representing the training and validation curves of accuracy and loss, respectively. Looking at Figure 4, it is possible to

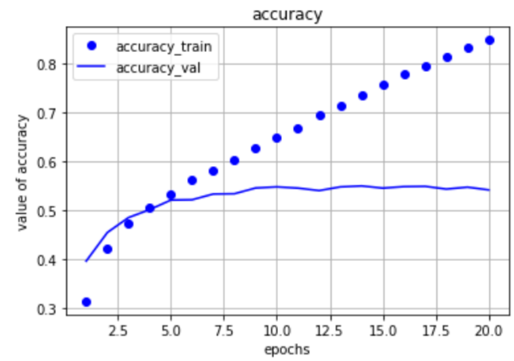

Figure 4: Plot of the accuracy function.

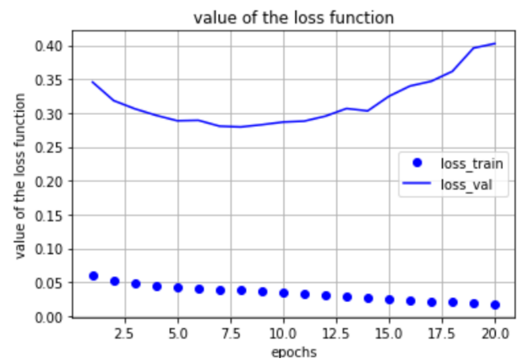

Figure 5: Plot of the loss function.

notice that the curve referring to accuracy validation begins to stabilise at 9 epochs, which suggests that instead of having trained the model with 20 epochs, 9 would have been enough. This can be corroborated by observing Figure 5 where, starting at the 9^{th} epoch, the training and loss validation curves begin to move further and further away.

With the help of these two charts, it is also possible to perceive a clear case of overfitting. This happens when the loss validation curve decreases to a certain point and then begins to increase again, which is visibly the case. However, several techniques have already been developed to prevent this problem from happening or at least to decrease it. One of the most effective methods is to use cross validation. Instead of dividing the data into 2 or 3 sets, cross validation divides it into several ones. It is intended that the model be trained in all sets, except for one at each step. So if we have x sets, the model is trained x times with a new test set at each step.

In Figure 6 we can see that the recognition rate of happiness was significantly higher than the others, followed by sadness, while the lowest is the expression of disgust, where we do not have any prediction at all.

The fact that there are no counts of the disgust emotion label in the test set of Figure 6 can be explained by the fact that in the dataset we have a large discrepancy between the test labels with this emotion and the other emotions (the second lowest test label count - the surprise emotion - has at least 150 more counts than the disgust labels). Similarly, the *CNN* model was trained with a training set containing the same

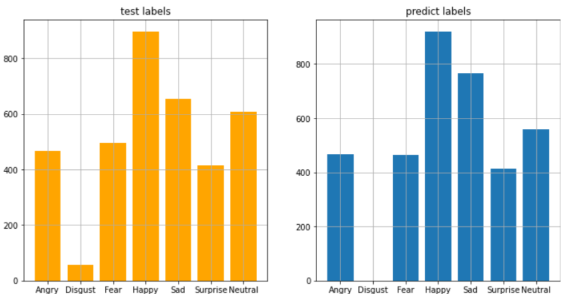

Figure 6: Comparison of the counts of the facial expressions labelled in the test model.

difference, resulting in the model being much less sensitive in identifying this emotion. In this way, we can say that the dataset could be composed of a more uniform spectrum of the emotions considered, allowing us to obtain a better trained model.

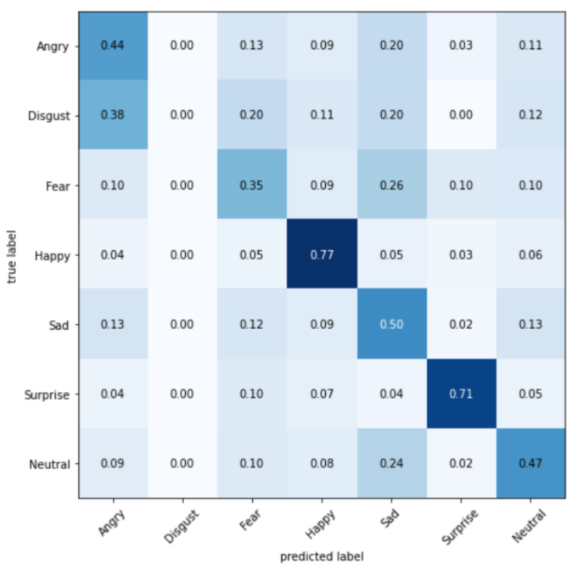

Figure 7: Confusion matrix of the model.

Finally, there is a confusion matrix. This is a technique used to summarise the performance of a sorting algorithm. Calculating it can give a better idea of what the rating model is hitting and what types of mistakes were made, thereby showing the ways in which the classification model gets confused when making predictions. Each row of the array corresponds to a predicted class and each column to a real class.

Looking at Figure 7 and focusing at the diagonal line of the medium (which concerns the percentage of correctly recognised emotions), it is possible to verify, as already mentioned above, that the type of emotion to which the best results are obtained, is the happy class, with 77% of certainty. On the other hand, the emotion to which the lowest results are obtained is disgust, because this model cannot classify any of the images of this class correctly. These results are expected, since the class that contains a smaller number of images is precisely the disgust class. The opposite happens with the happy class, which presents the largest number of images for training and consequent testing. Thus, to improve the performance of this classification model, more training data could be added to the classes in which worse results were obtained (disgust, fear, angry and neutral).

6 Conclusions

When someone has a physical illness or issue, for example they broke their foot and are in need of surgery, there are countless ways that the medical professionals can diagnose and assist them. However, to this day, the same logic and resources are not applied to mental health.

Discussing mental health is extremely important to not only have a better physical health but quality of life. It is essential to have a sensitive, empathetic and well educated conversation, which we hope that this paper will add.

The proposed model's ambition is to help solve the problem of emotion recognition through facial expressions. There is a simultaneous consideration of efficiency and accuracy in order to have a well rounded view on the effectiveness of the project.

By using the *CNN* model to detect eye and mouth placement as well as identifying the seven types of emotions considered (angry, disgust, fear, happy, sad, surprise and neutral) through the neural network method, the combination of efficiency and accuracy is achieved, which means that it can be applied to improve a mental illness diagnosis.

In fact, there has been published research on this subject. For example, one article uses the same convolutional network that is used in our study (*CNN*). However, comparing the accuracy results of both works, those of our colleagues are much better, presenting an accuracy of 98.5% [7]. After a detailed analysis of that study, it was possible to realise the probable existence of overfitting, since the dataset is quite small (contains only 20 images). Also, analysing the dataset used, one finds out that it only contains images of one person. Thus, even though the study presents a better accuracy than ours, our study is more adaptable to any type of person since the model was trained with images belonging to several people.

In another article we found, even though the authors use the same dataset as in our study, the model is trained based on *VGGNet* and not *CNN*. The calculated accuracy is much better (73.28%) than ours (55.2%) [13] so, for future work, the next considerable step should be to apply the *VGGNet* method in order to have a more extensive and well rounded comparison with other developed studies.

Due to the complexity and increased importance of facial recognition, there are other several studies conducted in this field for mapping emotions. There are a multitude of factors to take in consideration and our study does not come without its shortcomings. In fact, they must be pointed out so that others can improve on it.

The problem we encountered was the poor lighting of the images, which undoubtedly has an impact on the processing of the facial expression and thus, the final emotion recognition. Besides, when talking about emotion/facial recognition, there are a lot of societal factors to take in count.

It has been widely documented that the perception of facial expressions differs across cultures [33]. Because of this, the algorithm that is designed by humans will probably reflect their insufficient knowledge about other cultures, therefore having some biases.

As a matter of fact, this connects to another massive issue in the community: racial biases and discrimination in AI [34]. Studies show that the poorest accuracy in facial recognition systems is consistently found in subjects who are female, black and between 18 to 30 years old [35].

Despite the fact that our dataset is somewhat diverse, in order to curb this issue it is imperative to develop a dataset with a more diverse group of people that appropriately reflects the society we live in.

Acknowledgements

Some words of gratitude must be extended to our supervisor professor Luís Macedo for his unconditional support, expert advice and encouragement throughout the development of this project.

References

[1] "A brief history of facial recognition," accessed on 02.11.2021. [Online]. Available: https://www.nec.co.nz/market-leadership/publications-media/a-brief-history-of-facial-recognition/

[2] "What is mental illness," accessed on 04.12.2021. [Online]. Available: https://www.psychiatry.org/patients-families/what-is-mental-illness

[3] K. Molebatsi, K. Motlhatlhedi, and G. N. Wambua, "The validity and reliability of the patient health questionnaire-9 for screening depression in primary health care patients in botswana," *BMC Psychiatry*, vol. 20, p. 295, 2020. [Online]. Available: https://doi.org/10.1186/s12888-020-02719-5

[4] Y.-K. Kim, *Frontiers in Psychiatry: Artificial Intelligence, Precision Medicine, and Other Paradigm Shifts*. Springer, 2019.

[5] "What's the dsm?" accessed on 30.12.2021. [Online]. Available: https://psychcentral.com/lib/dsm-5#whats-the-dsm

[6] S. Gans, "An overview of the dsm-5," accessed on 30.12.2021. [Online]. Available: https://www.verywellhealth.com/an-overview-of-the-dsm-5-5197607

[7] S. Khan, M. H. Javed, E. Ahmed, S. A. A. Shah, and S. U. Ali, "Facial recognition using convolutional neural networks and implementation on smart glasses," in *2019 International Conference on Information Science and Communication Technology (ICISCT)*, 2019, pp. 1–6.

[8] J. Teuwen and N. Moriakov, "Chapter 20 - convolutional neural networks," in *Handbook of Medical Image Computing and Computer Assisted Intervention*, ser. The Elsevier and MICCAI Society Book Series, S. K. Zhou, D. Rueckert, and G. Fichtinger, Eds. Academic Press, 2020, pp. 481–501. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128161760000259

[9] J. Brownlee, "Overfitting and underfitting with machine learning algorithms," accessed on 31.12.2021. [Online]. Available: https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/

[10] Hargurjeet, "7 best techniques to improve the accuracy of cnn w/o overfitting," accessed on 31.12.2021. [Online]. Available: https://medium.com/mlearning-ai/7-best-techniques-to-improve-the-accuracy-of-cnn-w-o-overfitting-6db06467182f

[11] B. Han, J. Sim, and H. Adam, "Branchout: Regularization for online ensemble tracking with convolutional neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 521–530.

[12] N. A. S. Badrulhisham and N. N. A. Mangshor, "Emotion recognition using convolutional neural network (CNN)," *Journal of Physics: Conference Series*, vol. 1962, no. 1, p. 012040, jul 2021. [Online]. Available: https://doi.org/10.1088/1742-6596/1962/1/012040

[13] Y. Khaireddin and Z. Chen, "Facial emotion recognition: State of the art performance on fer2013," 2021.

[14] C. Pramerdorfer and M. Kampel, "Facial expression recognition using convolutional neural networks: State of the art," 12 2016.

[15] K. Liu, M. Zhang, and Z. Pan, "Facial expression recognition with cnn ensemble," 09 2016, pp. 163–166.

[16] F. Lardinois, M. Lynley, and J. Mannes, "Google is acquiring data science community kaggle," accessed on 29.12.2021. [Online]. Available: https://techcrunch.com/2017/03/07/google-is-acquiring-data-science-community-kaggle/

[17] P.-L. Carrier and A. Courville, "Challenges in representation learning: Facial expression recognition challenge," accessed on 06.11.2021. [Online]. Available: https://www.kaggle.com/drcapa/facial-expression-eda-cnn/notebook

[18] V. Tatan, "Understanding cnn (convolutional neural network)," accessed on 28.12.2021. [Online]. Available: https://towardsdatascience.com/understanding-cnn-convolutional-neural-network-69fd626ee7d4

[19] S. Saha, "A comprehensive guide to convolutional neural networks — the eli5 way," accessed on 29.12.2021. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[20] "About keras," accessed on 16.11.2021. [Online]. Available: https://keras.io/about/

[21] "An end-to-end open source machine learning platform," accessed on 29.12.2021. [Online]. Available: https://www.tensorflow.org/

[22] Pratik, “Numpy and pandas tutorial – data analysis with python,” accessed on 29.12.2021. [Online]. Available: https://cloudxlab.com/blog/numpy-pandas-introduction/

[23] “What is numpy?” accessed on 28.12.2021. [Online]. Available: https://numpy.org/doc/stable/user/whatisnumpy.html

[24] E. Team, “What is pandas in python?” accessed on 28.12.2021. [Online]. Available: https://www.educative.io/edpresso/what-is-pandas-in-python

[25] “What is colaboratory?” accessed on 29.12.2021. [Online]. Available: https://research.google.com/colaboratory/faq.html

[26] L. Soares, “A practical summary of matplotlib in 13 python snippets,” accessed on 29.12.2021. [Online]. Available: https://towardsdatascience.com/a-practical-summary-of-matplotlib-in-13-python-snippets-4d07f0011bdf

[27] “os — miscellaneous operating system interfaces,” accessed on 29.12.2021. [Online]. Available: https://docs.python.org/3/library/os.html

[28] “3.3. metrics and scoring: quantifying the quality of predictions,” accessed on 29.12.2021. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html

[29] E. Alizadeh, “Mlxtend: A library with interesting tools for data science tasks,” accessed on 29.12.2021. [Online]. Available: https://towardsdatascience.com/mlxtend-a-python-library-with-interesting-tools-for-data-science-tasks-d54c723f89cd

[30] “The model class,” accessed on 29.12.2021. [Online]. Available: https://keras.io/api/models/model/

[31] J. Brownlee, “Difference between a batch and an epoch in a neural network,” accessed on 29.12.2021. [Online]. Available: https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/

[32] S. Russel and P. Norvig, *Artificial Intelligence: A Modern Approach*, fourth edition ed. Pearson, 2021.

[33] R. E. Jack, R. Caldara, and P. G. Schyns, “Internal representations reveal cultural diversity in expectations of facial expressions of emotion.” *Journal of experimental psychology. General*, vol. 141, pp. 19–25, 2012.

[34] B. N. Reyes, S. C. Segal, and M. C. Moulson, “An investigation of the effect of race-based social categorization on adults’ recognition of emotion.” *PloS one*, vol. 13, 2018.

[35] A. Najibi, “Racial discrimination in face recognition technology,” accessed on 30.12.2021. [Online]. Available: https://sitn.hms.harvard.edu/flash/2020/racial-discrimination-in-face-recognition-technology/