CO3519 Assignment – Facial Emotion Recognition and Classification using Machine Learning

Author: James Birkenhead | G20983016

## ***Introduction***

Theory of Mind Artificial Intelligence (AI) encompasses the enablement of AI systems to understand human emotions. A Theory of Mind AI would have the capability to interpret human needs, emotions and behaviours and respond appropriately.

Whether current AI technologies such as Large Language Models (LLMs) have achieved Theory of Mind is contested, however a potentially powerful application of current AI capability is for Facial Expression Recognition (FER). The detection of emotions is typically based on the analysis of facial landmark positions such as nose, eyebrows, mouth etc. and changes to those positions can be analysed. These can then be classified to various emotions (European Data Protection Supervisor, 2021).

FER is deemed to be important since much communication is non-verbal, with some studies suggesting up to 60-80%. FER has numerous applications from areas such as education, neuroscience and psychology, to autopilot and more (Huang et al., 2023).

This paper will explore the implementation of a Machine Leaning (ML) algorithm to recognise and classify basic facial emotions, demonstrating the power of AI in this area.

## ***State Of the Literature***

FEATURE EXTRACTION – A COMPARISON OF LBP AND HOG

Local Binary Patterns (LBP) are a method of feature extraction which can be applied to facial expression recognition. Faces are processed to extract texture patterns by thresholding a 3x3 neighbourhood of each pixel with the centre pixel value and considering the result as a binary number. A histogram is then formed from these labels that can represent the unique textures of a face. The histogram can then be used to train a machine learning model to recognize various faces within the images. (Ghorbani, Targhi and Dehshibi, 2015)

Histogram of Oriented Gradients (HOG) is another method of feature extraction that works with histograms, however the process for feature extraction is slightly different. Occurrences are counted of edge orientations in a localized image neighbourhood. These neighbourhoods represent facial contours and textures that can be used to distinguish emotions.

Adouani, Ben Henia, and Lachiri (2019) compared the LBP and HOG methods alongside the Haar like features algorithm. A sequence of videos was taken from the multimodal DEAP database which contains several hundred videos of facial recordings which can be used for emotional analysis. A video sequence was then inputted, converted to grayscale, and the various facial detection techniques were tested, evaluated, and compared. The comparison was performed via a True Positive Rate which measured the proportion of faces correctly identified, and a False Negative Rate which measured the proportion of faces which yielded negative outcomes. In total, videos of seven human participants were used.

Julina and Sree Sharmila (2019) took a similar approach in evaluating the various algorithms by testing the feature based facial emotion recognition via a sequence of videos. Various frames from the video input were separated, and the faces were detected and extracted using the HOG and LBP techniques. This study also assessed classification, although this is out of the scope of this stage of the review. To measure the accuracy of the two models, the sum of true positives/negatives was divided by the sum of true positives/negatives and false positives/negatives.

A consensus that can be drawn from the literature is that HOG produces a higher level of accuracy compared to LBP. Adouani, Ben Henia, and Lachiri (2019) found that HOG was more accurate than LBP with a higher true positive rate and lower false negative rate across all 7 of the videos watched. Overall, HOG achieved a 92.68% detection rate and LBP was much inferior with a 32% reduction in correct detection compared to HOG. Likewise, Julina and Sree Sharmila (2019) also concluded that HOG achieved a higher level of accuracy, with HOG achieving an accuracy rating of 87% and LBP achieving an accuracy rating of 64%. While Adouani, Ben Henia, and Lachiri (2019) produced a higher magnitude of accuracy difference between HOG and LBP compared to Julina and Sree Sharmila (2019), this could be explained by differences in the test data across the two studies. However, the overall picture remains consistent and suggests that HOG should be chosen as the method to use for facial detection owing to its’ high accuracy compared to LBP.

EMOTION CLASSIFICATION – A COMPARISON OF SVM AND RF

Support Vector Machine (SVM) is a powerful machine learning algorithm which can be used for linear and nonlinear classification. SVM identifies an optimal hyperplane – a generalization in an N dimensional space - which is then used to separate various data points into several classes. With the placement of the hyperplane, SVM attempts to maximise the scale of the margin between the closest points of the various classes, known as the support vectors. This helps to improve the level of confidence in the resultant classifications.

Decision Trees are non-parametric supervised learning algorithms which can also be used for classification. Decision Trees consist of a hierarchical tree structure, with a root node that contains a set of branches feeding into internal decision nodes. Branches stemming from these nodes may lead to further internal decision nodes, however the algorithm will end at a leaf node which represents the outcome of the decision made. Since using a single decision tree could be prone to error and inaccuracy, the Random Forest (RF) technique can be used. This involves combining the predictions of multiple decision trees, improving the level accuracy and robustness within the model.

Kremic and Subasi (2016) assessed SVM and RF to inform decisions to be made for a mobile computer system which would detect and classify human faces. For their study, they acquired images from the International Burch University (IBU) face image dataset containing over 13000 images of 1680 people and conducted the following…

1. Read the image.
2. Detect the skin colour and convert to grayscale.
3. Create a histogram to extract features from the faces.
4. Classify the faces using SVM and RF
5. Authentication was carried out via verification and identification of the faces.

The accuracy of the algorithms was then compared. For SVM, this study also compared accuracy when changing several parameters.

* Changing the kernel function – in SVM the kernel is used to compute numerous decision boundaries in a feature space. Two different kernels were compared – a PUK kernel and a linear kernel.
  + A PUK kernel is a type of non-linear kernel, typically producing curved decision boundaries. Non-linear kernels allow for more complex decision boundaries such as curves, circles, and other more intricate shapes.
  + A linear kernel is used when the data is linearly separable. The linear kernel produces a linear hyperplane in the feature space.
* Changing the classifier (C) value controls the trade-off between the margin and misclassifications. Generally, a smaller value of C promotes a wider margin with a higher risk of misclassification, and a larger value of C promotes a smaller margin with a lower risk of misclassification.

Nugrahaeni and Mutijarsa (2016) also compared SVM and RF, and also alongside the K Nearest Neighbour (KNN) algorithm. The study used images from the Extended Cohn Kanade (CK+) database which contains 593 video sequences from 123 different subjects of various age, genders, and heritage. The training datasets were pre-processed, and faces were detected by locating the centre of the face and getting the x and y positions of main facial features. The distance of the x and y positions from the centre of the face were then stored in an ARFF file. This was then fed into a library and used to train the machine learning algorithms, resulting in numerous evaluation metrics such as accuracy, precision, F-measure, and confusion matrix.

In terms of accuracy, the picture is quite mixed across the literature with some conflict. Kremic and Subasi (2016) found that the level of accuracy between the two algorithms was remarkably similar, with RF achieving an accuracy rating of 97.17% and SVM achieving a maximum accuracy of 97.94%. A caveat in this study is that the SVM figure only produced an accuracy rating of 97.17% under one condition where the value of the classifier was 100 and using a linear kernel. However, the range of accuracy ratings was still small with the lowest accuracy rating being 95.89% which would only be inferior to RF by 2.05%. This concludes that RF is the slightly favourable algorithm over SVM in terms of accuracy, although the difference is very small. However, Nugrahaeni and Mutijarsa (2016) found that for a small amount of training data, SVM outperformed RF with accuracy figures of 80% and 76.97% respectively, although if the amount of training data increased, then this reversed with RF producing a 98.85% accuracy rating and SVM producing 90%. A potential explanation for the discrepancy could be related to the datasets used. A limitation of both studies is that they did not specify which images they used, or in the case of Kremic and Subasi (2016), how many or which images they used from the IBU dataset. This means there could have been potential bias in the images used for training and testing such as sex and ethnicity. However, since this limitation is shared across the papers, a conclusion that can be drawn from this is that RF has the highest potential accuracy out of the two algorithms, however this will require a larger amount of training data to be significantly stronger than SVM. If the amount of training data available is small, then SVM may be preferable with the appropriate configuration used by Kremic and Subasi (2016), since it was able to produce similar results to RF. In the context of this study, since the combination of the JAFFE dataset and the CK+ dataset (See Datasets section) will lead to a large amount of training data, the indication is that RF will be preferable.

## ***Datasets***

The JAFFE dataset is a set of 213 images from 10 Japanese female subjects with various facial expressions. This dataset was created for non-commercial scientific research. Each of the ten subjects did seven facial expressions to correspond to the following emotions…

* Happiness
* Sadness
* Surprise
* Anger
* Disgust
* Fear
* Neutral

A sample of these images can be found in Appendix A.

The images are already in grayscale format and clearly show a face. Converting images to grayscale is important for this study since some detection algorithms like HOG are defined based on grayscale values, and generally using grayscale also reduces complexity which in turn reduces the demands on memory and processing power. There is also no evidence of significant background noise in the sample images that would need to be filtered out. A link to the dataset can be found here <https://paperswithcode.com/dataset/jaffe>. Before being inputted into the model, the various images were organised into their respective emotions and grouped together into folders.

The Cohn-Kanade Dataset (CK+) is a dataset that was created in 2000 for the purpose of promoting research into automatically detecting facial expressions. There are 593 video sequences from 123 different subjects ranging from 18 to 50 years old with various genders and heritage. This is advantageous compared to the JAFFE dataset since it only focuses on Japanese female subjects, meaning the model can be trained on a more diverse range of subjects. Once again, there is no evidence of significant background noise in the sample images that would need to be filtered out. A link to the dataset can be found here <https://www.kaggle.com/datasets/shuvoalok/ck-dataset?resource=download>.

For all images from both datasets, the images will be resized to 64x64 pixels before the HOG feature extraction is conducted. This will ensure that there is consistency in the size of the images being used for feature extraction.

## ***Model Development***

A diagram of the proposed model can be found in Appendix B. The stages are as follows…

**Input grayscale image.**

Images will be input from the dataset. Each of these images are grayscale since HOG expects grayscale values. Grayscale also reduces complexity which in turn reduces the demands on memory and processing power.

**Facial detection using HAAR Cascade**

Haar cascade is an algorithm which can be used to detect objects in images. This algorithm is considered to be powerful for facial detection because it can be used to detect various facial features such as eyes, nose, and mouth (Mittal, 2020).

The algorithm will detect faces by creating a HAAR cascade classifier using a set of positive and negative images, the positive images containing a face. The classifier will then be used to scan each of the images for faces. If the features consistent with a face can be detected, then the image will be added to the list of images to be processed for feature extraction and emotion recognition. If these features cannot be detected, the algorithm will conclude that the image is not a face, and it will be rejected by the model.

**Feature Extraction using HOG.**

Once a face has been detected on an image, Feature Extraction will be conducted on it using the Histogram of Oriented Gradients (HOG) method. This is a popular method for feature extraction which has been chosen because the literature concludes that HOG produces a higher level of accuracy than other feature extraction methods such as LBP.

This phase will involve decomposing each image into a dense array of cells and calculating gradients for each cell. This is done by calculating the difference of pixel values in the x and y directions and then using Pythagoras theorem to determine the total magnitude. Gradients are computed for all of the pixels in a cell, and then the gradients are then collated to form a histogram, which contains various features of the image. The gradients are especially important since they will contain much more information than flat regions which can be used to determine facial features.

Finally, the gradients are normalised because localised image gradients are quite sensitive to overall lighting. This will help to improve accuracy. Groups of cells are grouped together into blocks, and a normalized vector for each block is calculated by getting the root sum of the squares of all the block features and dividing each block feature by this value.

Once these features have been extracted from the face, then they can be fed into a machine learning model to classify emotions.

**Emotion Classification using Random Forest (RF)**

Once the facial features have been extracted, the emotions will be classified using Random Forest (RF) since the literature indicates that RF is at least slightly more accurate than alternatives such as SVM. RF is based on the concept of decision trees – a supervised learning algorithm consisting of a hierarchical tree structure, with a root node that contains a set of branches feeding into internal decision nodes. Branches stemming from these nodes may lead to further internal decision nodes, however the algorithm will end at a leaf node which represents the outcome of the decision made.

RF expands on this concept by using multiple decision trees, which improves accuracy and reduces error by avoiding a single point of failure that comes with a single decision tree. This will be implemented using a Decision Tree Classifier class in Python.

The classifier will be trained on a set of training data to allow for various emotions to be recognised. A confusion matrix will also be produced. The confusion matrix is a table with several combinations of predicted and actual values, with the cells showing the number of correctly classified faces and incorrectly classified faces. This will provide the user with further insight and information about the accuracy of the model.

Once the training has been completed, test images will be passed into the classifier which will predict the emotion being displayed.

## ***Model Evaluation***

The model was trained and tested using the two datasets proposed. Each dataset contained a set of images for training the model, and then a set of images for testing the model.

The proposed model was implemented using Python. Various libraries were used to facilitate the model including…

* OpenCV for Haar Cascade.
* numpy.
* Matplotlib to render HOG features.
* Sklearn.tree for the Decision Tree Classifier.
* Skimage.feature for the HOG capability.
* Sklearn.metrics for the accuracy score and confusion matrix used to measure the accuracy of the model.
* Seaborn to generate a Confusion Matrix.

The model was trained on six separate emotions…

* Anger
* Fear
* Happiness
* Neutral
* Sadness
* Surprise

The first round of training was carried out using a combination of the JAFFE dataset and the CK+ dataset.

Following the training process, the model was tested using 198 images categorised into the same emotions using for training. A Confusion Matrix was then generated which gave a visual insight into the performance of the Machine Learning algorithm. This matrix displayed the number of instances of each class of emotion produced by the model.

In a Confusion Matrix, the diagonal elements represent the number of correct predictions for each of the emotion classes. Misclassifications on the other hand are represented by the number of elements off the diagonal line. Therefore, a higher number of diagonal elements indicates that the accuracy of the model is higher (GeeksForGeeks, 2018).

Several key figures could then be deduced from the matrix.

* **True Positives (TP)** measures the number of emotions correctly predicted by the model.
* **True Negatives (TN)** measures the number of times a different emotion was correctly predicted by the model.
* **False Positive Rate (FP)** measures the number of times the model incorrectly predicted an emotion when the true emotion was different.
* **False Negatives (FR)** measures the number of times the model failed to predict an emotion when it was the true emotion.

Once the figures for the above values were obtained, several metrics could be collected to describe the model’s performance.

* **Accuracy** is calculated by dividing the sum of True Positives and True Negatives by the Total number of True and False Positives and True and False Negatives.
* **Precision** measures how accurate the model’s positive predictions are, calculated by dividing the number of True Positives by the sum of True and False Positives.
* **Recall** measures the number of true positives divided by the count of actual positive outcomes (true positives + false negatives). This can be used to determine how well the model can identify the real true result.
* **F1 Score** measures the harmonic mean between precision and recall. This is considered a strong metric to measure the overall performance of the classification model.

(GeeksForGeeks, 2018)

The Confusion Matrix generated on the initial training run can be found in Appendix C.

From this matrix, the TP, FP, TN, and FN were calculated alongside the accuracy, precision, recall and F1 scores. The table containing this data can be found in Appendix D.

From this data, several conclusions can be drawn…

* **The model performed best on classifying Happiness and Surprise** – The model’s performance in classifying happiness and surprise was significantly greater than the other emotions. The model boasted the joint second and third highest accuracy scores on happiness and surprise, respectively. Happiness and surprise also boasted the two highest precisions, recall and F1 scores.
* **Accuracy was strong overall** – Despite variance in Precision, Recall and F1 scores, the accuracy of the model was strong overall.
* **The model performed weakest on classifying Anger despite having the joint second highest Accuracy Score in this area** – Despite an accuracy score of 83% for anger, the precision score was second lowest at 27%, the recall score was the lowest at 15% and the F1 score was the lowest at 19%, indicating that the model’s overall performance was lowest at classifying anger. The juxtaposition between accuracy and overall performance in this category is notable and warrants investigation.
* **Sadness did not fare much better** – The F1 score for classifying sadness was only 2% higher than anger, and the accuracy of classifying sadness was 6% less.

A notable contrast could be observed at times between accuracy and other metrics leading to a low F1-Score. This could be observed particularly on Sadness and Anger, and Fear to a lesser extent. Several potential factors could explain this…

* **Underfitting** – In Machine Learning, underfitting occurs when the model is too simple and therefore fails to capture at least some patterns in the data. This could be the case with this model since Random Forest consists of a set of decision trees, and decision trees are relatively simple with binary outcomes. There are also more advanced methods of extracting facial features than HOG, such as Convolutional Neural Networks (CNNs). These use a series of layers, each of which detects various features of an input image. Each layer builds on the output of the previous layer to recognize very detailed patterns (Awati, 2022). Since HOG and RF are simpler, underfitting could have potentially occurred especially when determining the features that distinguish sadness, anger, and fear, leading to poor performance on the testing data in these areas, while the performance in other areas was much stronger.
* **Class Imbalance within training dataset** – More images were stored within the training dataset for surprise, happiness, and neutral emotions than fear, anger, and sadness, which could explain the higher performance.
* **Ethnic diversity causing conflict in training dataset** – The JAFFE dataset focuses specifically on women from an Asian/Japanese origin whereas the CK+ dataset focuses on various other ethnicities. The varying ethnicities could’ve potentially caused conflict when training the model since the various features could have been different.

To experiment with changing the dataset, the model was re-run with an updated dataset where the number of training images was kept the same across each emotion classification. The updated performance results can be found in Appendix E.

Making this change meant that there was less variance in F1 scores across all the various emotions. The precision and recall scores of anger and sadness improved alongside the F1 scores. However, the F1 score of fear decreased further, alongside its’ accuracy. The recall score improved however the precision score decreased. This indicates that there were potentially more factors playing a role in the poor performance in specifically classifying fear than merely class balance.

A conclusion that could be drawn from this is that the contrasts between the performance of the model on happiness and surprise and performance on fear, anger and sadness could be explained by the presence of more training data for the former two emotions, so increasing the amount of training data could improve the performance of the model further. A bar chart comparing the F1 scores of the model on the two datasets can be found in Appendix F. While this may improve wider performance, classifying some emotions like fear remains challenging for the model, indicating limitations.

As an additional experiment, the trial was run again using the CK+ dataset and JAFFE dataset in isolation rather than combined. The results of these test runs can be found in Appendix G and H. A bar chart visualizing the differences between CK+ and JAFFE can be found in Appendix I.

The model generally performed better on the CK+ dataset, with a much higher F1 score for happiness, and higher scores for neutral, sadness and surprise. However, the model performed more poorly on CK+ for fear and happiness, and fear could not achieve a true positive. Since performance was universally weaker for categorising fear and anger across the two datasets, this indicates a common failure mode and limitation of the model in general rather than a problem with the training data.

## ***Demonstration***

A demonstration video can be found here - [AI Assignment Implementation Demo.mp4](https://msuclanac-my.sharepoint.com/:v:/g/personal/jbirkenhead1_uclan_ac_uk/EZUAszBUV_RPr2NXJue1tVIBu4Y-nT4ssBRY0RhrRdxHKw?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJPbmVEcml2ZUZvckJ1c2luZXNzIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXciLCJyZWZlcnJhbFZpZXciOiJNeUZpbGVzTGlua0NvcHkifX0&e=Xgg2jd)

The source code itself along with the training and testing datasets used can be found at <https://github.com/JamesBirky15/AI-Assignment> in the “Code Implementation” folder.

## ***Conclusion***

In conclusion, while Facial Emotion Recognition is deemed to be a key area where AI has significant potential and benefit for society, establishing a robust and reliable method for this is deemed to be challenging. This study has proposed a method for Facial Expression Recognition based on the HOG method for Feature Extraction and Random Forest method for emotion classification. By completing this study, I have learned about the creation machine learning models for facial emotion recognition and classification. By implementing a machine learning model, I have become more aware of various approaches for feature extraction and taking those features and classifying them to various emotions. While the model was powerful in many ways, its’ simplicity also limited its’ potential since it seemingly underperformed in classifying more complex emotion such as fear. In the future, more complex algorithms could be explored such as Convolutional Neural Networks (CNNs). These use a series of layers, each of which detects various features of an input image. Each layer builds on the output of the previous layer to recognize very detailed patterns, making them more powerful than some of the methods discussed in this study (Awati, 2022).

## ***Appendix***

**Appendix A:** Sample of faces from JAFFE dataset (Lyons, Miyuki Kamachi and Jiro Gyoba, 1998)A collage of different facial expressions

Description automatically generated

**Appendix B:** Proposed Model



A screenshot of a computer game

Description automatically generated**Appendix C:** Confusion matrix generated by model.

**Appendix D:** Scores from first model run with combination of CK+ dataset and JAFFE dataset.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | TP | FP | TN | FN | Accuracy | Precision | Recall | F1 |
| Anger | 4 | 11 | 160 | 23 | 83% | 27% | 15% | 19% |
| Fear | 6 | 8 | 173 | 11 | 90% | 43% | 35% | 39% |
| Happiness | 23 | 8 | 151 | 16 | 88% | 74% | 59% | 66% |
| Neutral | 23 | 40 | 113 | 22 | 69% | 37% | 51% | 43% |
| Sadness | 6 | 25 | 147 | 20 | 77% | 19% | 23% | 21% |
| Surprise | 27 | 17 | 137 | 17 | 83% | 61% | 61% | 61% |

**Appendix E:** Scores from second model run where sample sizes for each emotion were made the same.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | TP | FP | TN | FN | Accuracy | Precision | Recall | F1 |
| Anger | 17 | 26 | 145 | 10 | 81% | 40% | 63% | 50% |
| Fear | 1 | 14 | 167 | 10 | 88% | 33% | 41% | 34% |
| Happiness | 20 | 4 | 155 | 19 | 88% | 83% | 51% | 63% |
| Neutral | 12 | 23 | 130 | 33 | 71% | 34% | 27% | 30% |
| Sadness | 8 | 25 | 147 | 18 | 78% | 24% | 30% | 27% |
| Surprise | 20 | 22 | 132 | 24 | 83% | 61% | 61% | 61% |

**Appendix F:** Comparison of F1 scores between original dataset and changed dataset where sample sizes for each emotion were the same.

**Appendix G:** Results for model run using JAFFE dataset only.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | TP | FP | TN | FN | Accuracy | Precision | Recall | F1 |
| Anger | 4 | 9 | 37 | 5 | 74% | 31% | 44% | 37% |
| Fear | 2 | 1 | 44 | 8 | 84% | 67% | 20% | 30% |
| Happiness | 1 | 2 | 44 | 8 | 82% | 34% | 11% | 17% |
| Neutral | 6 | 18 | 28 | 3 | 62% | 25% | 67% | 36% |
| Sadness | 1 | 8 | 38 | 8 | 70% | 11% | 11% | 11% |
| Surprise | 3 | 0 | 46 | 6 | 89% | 100% | 33% | 50% |

**Appendix H:** Results for model run using CK+ dataset only.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | TP | FP | TN | FN | Accuracy | Precision | Recall | F1 |
| Anger | 3 | 6 | 119 | 15 | 85% | 33% | 17% | 22% |
| Fear | 0 | 1 | 135 | 7 | 94% | 0% | 0% | N/A |
| Happiness | 21 | 13 | 100 | 9 | 85% | 62% | 70% | 66% |
| Neutral | 18 | 37 | 70 | 18 | 62% | 33% | 50% | 40% |
| Sadness | 3 | 15 | 111 | 14 | 80% | 17% | 18% | 17% |
| Surprise | 17 | 9 | 99 | 18 | 81% | 65% | 49% | 56% |

**Appendix I:** Comparison of F1 scores between model runs using JAFFE only and CK+ only.

A graph of different colored bars

Description automatically generated with medium confidence

## ***References***

Adouani, A., Ben Henia, W.M. and Lachiri, Z. (2019). *Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences*. [online] IEEE Xplore. doi:https://doi.org/10.1109/SSD.2019.8893214.

Awati, R. (2022). *What is convolutional neural network? - Definition from WhatIs.com*. [online] SearchEnterpriseAI. Available at: https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network.

European Data Protection Supervisor (2021). *Facial Emotion Recognition*. [online] Available at: <https://www.edps.europa.eu/system/files/2021-05/21-05-26_techdispatch-facial-emotion-recognition_ref_en.pdf>.

GeeksForGeeks (2018). *Confusion Matrix in Machine Learning - GeeksforGeeks*. [online] GeeksforGeeks. Available at: https://www.geeksforgeeks.org/confusion-matrix-machine-learning/.

Ghorbani, M., Targhi, A.T. and Dehshibi, M.M. (2015). HOG and LBP: Towards a robust face recognition system. *2015 Tenth International Conference on Digital Information Management (ICDIM)*. doi:https://doi.org/10.1109/icdim.2015.7381860.

Huang, Z.-Y., Chiang, C.-C., Chen, J.-H., Chen, Y.-C., Chung, H.-L., Cai, Y.-P. and Hsu, H.-C. (2023). A study on computer vision for facial emotion recognition. *Nature : Scientific Reports*, 13(1). doi:https://doi.org/10.1038/s41598-023-35446-4.

J. Kulandai Josephine Julina and T. Sree Sharmila (2019). Facial Emotion Recognition in Videos using HOG and LBP. *2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*. doi:https://doi.org/10.1109/rteict46194.2019.9016766.

Kremic, E. and Subasi, A., 2016. Performance of random forest and SVM in face recognition. *Int. Arab J. Inf. Technol.*, *13*(2), pp.287-293.

Lyons, M.J., Miyuki Kamachi and Jiro Gyoba (1998). The Japanese Female Facial Expression (JAFFE) Dataset. doi:https://doi.org/10.5281/zenodo.3451524.

Mittal, A. (2020). *Haar Cascades, Explained*. [online] Medium. Available at: https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d.

Nugrahaeni, R.A. and Mutijarsa, K. (2016). *Comparative analysis of machine learning KNN, SVM, and random forests algorithm for facial expression classification*. [online] IEEE Xplore. doi:https://doi.org/10.1109/ISEMANTIC.2016.7873831.