



BEN-GURION UNIVERSITY OF THE NEGEV
FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF INDUSTRIAL ENGINEERING AND MANAGEMENT

Face Emotions Recognition

Image Processing and Analysis Course

001-2902-31

Dr. Tarin Paz-Kagan

28.04.2024

Ofir Azulay • Moshiko Cohen • Peleg Eliyahou

[GitHub Link](#)

Table of Contents:

1	Introduction	2
1.1	Scientific Background	2
1.1.1	Emotion Recognition	2
1.1.2	Face Detection	2
1.1.3	Image Enhancement	3
1.1.4	Convolutional Neural Networks	3
1.2	Research Question	4
2	Methodology	4
2.1	Database Description	4
2.1.1	Data Acquisition	4
2.1.2	Data Characteristics.....	4
2.1.3	Examples of The Data.....	4
2.2	Facial Emotions Classification Pipeline.....	5
2.2.1	Face Detection	5
2.2.2	Pre-processing	5
2.2.3	Classification Model.....	6
3	Results	7
3.1	Models Evaluations.....	7
3.2	Selected Model	8
4	Conclusions and Discussion.....	9
5	Further Work.....	9
6	Bibliography	11
7	Appendices.....	12

1 Introduction

1.1 Scientific Background

In the last years, face expression recognition has become a popular area of research in computer vision which is applied in a wide variety of areas, specifically for human and computer interaction. The recognition of facial expressions is not an easy problem, and the performances of face expression recognition systems depends on many conditions [1] such as performing pre-processing and the selection and training the classification model.

1.1.1 Emotion Recognition

Emotion recognition refers to the ability of computer systems to recognize and interpret human emotions expressed in images or videos. This technology aims to identify and analyze facial expressions, and other visual cues to infer the emotional state of people in an image. It is applied in various fields such as human-computer interaction, healthcare, market research, and entertainment. This technology enables systems to adjust their responses based on the emotional state of users, facilitating personalized experiences and providing valuable insights into human behaviour.

Many studies have proposed automatic emotion recognition, primarily utilizing a machine learning approach [2]. By extracting facial features from images, the system classifies the emotional state of individuals into predefined categories such as happiness, sadness, anger, surprise, fear, or neutrality. This classification is often performed using algorithms trained on labeled datasets, such as CNN.

Face Emotion Recognition usually involves pre-processing methods including face recognition and image enhancement.

1.1.2 Face Detection

Facial detection within the image processing field represents an advanced facet of object detection, focusing on the identification of human faces within digital images or video [3]. This technology leverages computer vision and machine learning algorithms to detect human faces and analyze facial characteristics. Facial detection refers to identifying all faces within an image and distinguishing them from the background and other objects. The approach to facial detection tasks varies based on the problem and project objectives.

Commonly utilized algorithms for facial detection include sophisticated deep learning models such as MTCNN (Multi-task Cascaded Convolutional Networks), HOG + Linear SVM and Haar cascades. **Haar cascades** are a technique used in object detection, particularly popularized for face detection. They involve applying a series of progressively more complex patterns to an image to detect the presence of a particular object or feature [4]. These patterns, known as Haar-like features, are rectangular regions that are contrastingly lighter or darker than their surrounding areas. The cascade aspect refers to the sequential application of these classifiers, where the algorithm quickly dismisses regions of the image that do not match

the patterns, thereby reducing computation time. Haar cascades have been widely utilized due to their effectiveness and efficiency in detecting objects within images.

1.1.3 Image Enhancement

Image enhancement refers to the process of improving the quality and information content of original data. Enhancement techniques are used to bring out specific features, increase the visibility of certain areas and correct common issues such as blurring or noise. The goal is to prepare the image for further analysis and computer vision tasks where improved clarity can aid in object detection and pattern recognition.

Pre-processing image enhancement steps are essential in facial and emotion recognition tasks [5]. Images often contain more information than necessary for the processing and the required task. To resolve unnecessary information and remove background noise, preprocessing algorithms such as Linear and Non-Linear Filters are usually applied [6]. In addition, edge detection algorithms are usually applied during the preprocessing phase to highlight frequent facial components [7].

1.1.4 Convolutional Neural Networks

Computer vision is a classic deep learning task. Deep learning is an implementation of artificial neural networks (ANN) with multiple hidden layers designed to mimic the functions of the human cerebral cortex [8]. Over the last few decades, ANN has been considered one of the most powerful tools and has become very popular in the literature due to its capability to handle large amounts of data [9]. In recent years, significant strides have been made in computer vision, with the development of classifiers for computer vision tasks on various datasets using various deep learning algorithms.

One of the most popular types of artificial neural networks is the Convolutional Neural Network (CNN), which has emerged as the dominant approach in deep learning for visual object recognition. CNN is a powerful algorithm capable of handling millions of parameters and demonstrating excellent performance in deep learning problems, especially those involving image data and pattern recognition, such as image classification tasks [8]. CNNs derive their name from a mathematical linear operation between matrixes called convolution. They take input images and convolve them with filters or kernels to extract features. An image is convolved with a $f \times f$ filter through a convolution operation. The window slides after each operation, and the features are learned through feature maps that utilize weights and biases [8]. CNNs consist of multiple layers, including convolutional layers, non-linearity layers, pooling layers, and fully connected layers [9].

During training, CNNs optimize their internal parameters using labeled data to minimize prediction errors. In inference, the trained CNN can classify new images by computing a probability distribution over possible classes and selecting the class with the highest probability as the predicted label. This approach enables CNNs to achieve high accuracy in image recognition tasks, making them particularly effective for tasks like facial expression analysis [10].

1.2 Research Question

In this project, we have decided to explore the task of image emotion classification, in particular, happy, angry and sad – three very common emotions. Understanding and identifying those emotions is a crucial aspect in many fields, such as human-robots interaction, and effectively classifying them can improve those interfaces.

Another aspect of image processing, is using different image processing techniques that could, hopefully, help the model to better understand the different classes' patterns, and improve the performances of the learning models in classifying the emotions expressed in the images.

This leads us to our project's research question:

Can we use image processing techniques for the task of image emotion classification, and whether there are preprocessing techniques that will improve the classification?

2 Methodology

2.1 Database Description

To answer our project's research question, we have collected a dataset containing images expressing emotions of the three classes: happy, angry, and sad.

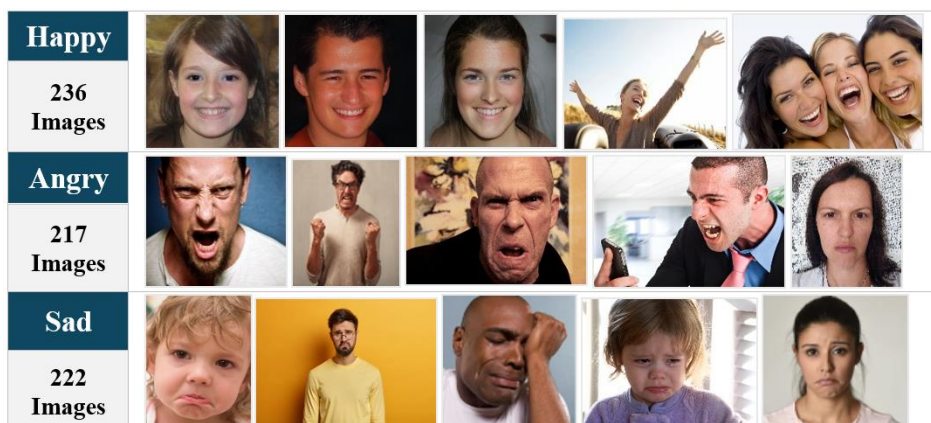
2.1.1 Data Acquisition

We combined a few open-source datasets available online, to achieve diverse and well-represented images for each of those emotions. Some of those datasets are Kaggle's "Human Face Emotions", "Facial Emotion Recognition Dataset", "STOCK2FER" and "Young AffectNet HQ" datasets. [[Appendix 1](#)]

2.1.2 Data Characteristics

After combining the datasets, removing problematic images and balancing between images expressing different emotions, we achieved a dataset of 675 images, with a good balance between classes, as there were 217 images expressing anger, 236 images expressing happiness and 222 images expressing sadness. The images were in JPG, JPEG and PNG format, from 10KB to 10,291KB.

2.1.3 Examples of The Data



2.2 Facial Emotions Classification Pipeline

The project was implemented with few main phases:

First, we performed a face detection algorithm over the images, and split our dataset into train, validation, and test sets. Next, we performed on each image all the examined pre-processing techniques, and built a CNN model for each preprocessing. We trained the models over the train set, validated over the validation set and predicted the emotions classification on images at the test set. As the final step, we evaluated the models' performances and chose the optimal model, according to our evaluation metrics.



2.2.1 Face Detection

After we gathered the classes images, we performed a face detection algorithm for detecting the different faces in the images. This will make sure that we will be able to classify emotions of images with multiple individuals faces. The face detection was performed using OpenCV's Haar cascades algorithm, one of the most popular OpenCV's face detection algorithms, first introduced by Viola and Jones in their 2001 publication [11]. One of the main advantages of this algorithm is its speed, while it rapidly detects the faces in the images, which was important to us due to our limited computational resources. Also, we got familiar with this algorithm in class, and we wanted to implement and practice our acquired knowledge, and use this course learned material.

Each image has been face detected by OpenCV's cascade classifier, and we saved the detected faces for further steps. As stated, some of the images contained more than one individual, and each of them was potentially be detected, resulting in few images with multiple faces detection. Another step we performed is manually removing wrong detected faces. There were a few of them, and we decided to remove them as we do not want our model to be trained or tested over non-face images, as they cannot express any emotion. After this face detection step, we were still maintaining the classes balance with 202 angry faces (~37%), 197 happy faces (~36%) and 142 sad faces (~26%). [[Appendix 2](#)]

2.2.2 Pre-processing

Next, we have applied several different preprocessing techniques, to study if any of them will improve the classification. We examined the following methods: [[Appendix 3](#)]

1. **Without Preprocessing** – This “preprocessing” left the detected faces as they were originally.
2. **Low Pass Filters** – Emotions classification often relies on extracting features from facial images, such as facial contours, textures, and landmarks. All the three below low-pass filters can help smooth out irregularities in these features, making them more consistent and easier to analyze. Also, it can reduce the irrelevant details in the image.

- 2.1. Gaussian Blur – Applying gaussian blur with different kernel sizes of (3,5,7). This was applied to each channel separately, and then merging the channels. This low-pass filter averages each pixel's value with its surrounding pixels weighted by a Gaussian distribution.
- 2.2. Median Blur – Applying median blur with different kernel sizes of (3,5,7). This was applied to each channel separately, and then merging the channels. This low-pass filter replaces each pixel's value with the median value of its neighborhood within the given kernel.
- 2.3. Mean Blur – Applying mean blur with different kernel sizes of (3,5,7). This was applied to each channel separately, and then merging the channels. This low-pass filter replaces each pixel's value with the mean value of its neighborhood within the given kernel.
- 2.4. Filter2D – Applying 2D low-pass filters with the kernels:

$$np.array([[0,1,-1,0],[1,x,-x,-1],[1,x,-x,-1],[0,1,-1,0]], np.float32)$$
, for x 's 3,5 and 7.
 This low-pass filter performs a 2-dimensional convolution operation, which involves sliding a kernel matrix over the image and computing the weighted sum of pixel values in the neighborhood defined by the kernel at each location.
3. Contrast Enhancement – Applying contrast enhancement by histogram equalization over each channel separately, and then merging the channels. This will give a linear trend to the cumulative probability function associated with the image channels. By enhancing the contrast of facial images, important features such as facial contours, wrinkles, and expressions can become more pronounced, making them easier to detect and analyze.
4. Thresholding – Applying thresholding of both dark and bright pixels over any channel separately, and then merging the channels. In our emotion classification task, we assume that some emotions are associated with brighter or darker regions, for example, happiness might be associated with a bright smile, and anger with dark expression wrinkles, and the thresholding might help in emphasizing them.
5. Canny – Applying Canny edge detection algorithm. By detecting edges in facial images using Canny edge detection, we can extract important features such as the shape of the eyes, eyebrows, mouth, and other facial landmarks that contribute to emotional expression. Also, it provides a simplified representation of the original image, which can help reduce the complexity of the input data for emotion classification algorithms, making the classification process more efficient and robust.
6. Merge Preprocessing – A manually built process where the image goes through the above preprocessing techniques. To avoid extra blurring, the only low-pass filter technique used in this step is gaussian blur with kernel size 3.

2.2.3 Classification Model

Architecture:

For each preprocessing technique, we built a convolutional neural network. The model starts with a convolutional layer with 10 filters of size (3,3), followed by another convolutional layer with the same configuration. ReLU activation functions are applied to introduce non-linearity.

After each pair of convolutional layers, a max-pooling layer with a pool size of (2,2) is added to reduce spatial dimensions and retain the most important features. This pattern of convolutional layers followed by max-pooling layers is repeated multiple times to gradually extract higher-level features from the input image. Towards the end of the network, there's a flattening layer to convert the multidimensional feature maps into a one-dimensional vector, which is then passed to a fully connected layer with 128 neurons and ReLU activation. To prevent overfitting, a dropout layer with a dropout rate of 0.45 is added before the final output layer. The output layer consists of 3 neurons with softmax activation, representing the probabilities of the input image belonging to each of the 3 classes.

Learning and Evaluation:

Each image went through the current preprocessing technique, and then the trained model was evaluated over the validation set and tested over the training set. The split of the train, validation and test sets was performed in advance, so we trained, validated, and tested all the different models over the same images every time. This ended up with 17 models (the combinations of preprocessing techniques and their parameters described above) which were evaluated by their accuracy, precision and recall.

3 Results

3.1 Models Evaluations

After we created 17 models, each of which was trained on a different pre-processed set of images, we examined their results by the three common performances metrics: Accuracy, Precision and Recall.

- Accuracy – This evaluation metric measures how often a machine learning model correctly predicts the outcome. It is calculated by dividing the number of correct predictions by the total number of predictions. In other words, accuracy answers the question: how often the model is right?
- Precision – This evaluation metric in multi-class classification evaluates the accuracy of positive predictions for each class individually. It quantifies the proportion of true positives among all images predicted as belonging to a specific emotion class. Higher precision indicates fewer false positives and better performance in correctly identifying images expressing the particular emotion. We calculated the precision of each class separately, and then calculated the averaged precision over all the classes to get the overall precision.
- Recall – This evaluation metric in multi-class classification evaluates the ability of a model to correctly identify all relevant instances of each class. It measures the proportion of true positives predicted for a specific emotion class out of all instances belonging to that class. Higher recall indicates fewer false negatives and better performance in capturing all instances of a particular emotion. Similar to precision, recall is calculated separately for each class and then averaged over all classes to obtain the overall recall. This metric is crucial in evaluating the model's effectiveness in comprehensively identifying instances across all classes.

The models' evaluations are given in [[Appendix 4](#)]. In general, the results were quite satisfactory, with measure values averaging above 70%. This suggests that the model effectively classified images into the three classes (without the model, the metrics values for each class classification should typically remain around 33%)

All preprocessing techniques, except for Canny, yielded improvements in the model's performance, effectively classifying facial expression images into their appropriate classes. The edge detection approach employed by Canny, which emphasizes the general contours of the image, might unintentionally overlook essential details necessary for precise facial emotions classification, making it unsuitable for this task.

Moreover, the overall the outcomes for the low pass filters preprocesses exceed the other methods performances. This suggests that blurring the image may enhance key facial expression areas and facilitate the extraction of features that help the model accurately classify images into emotion classes.

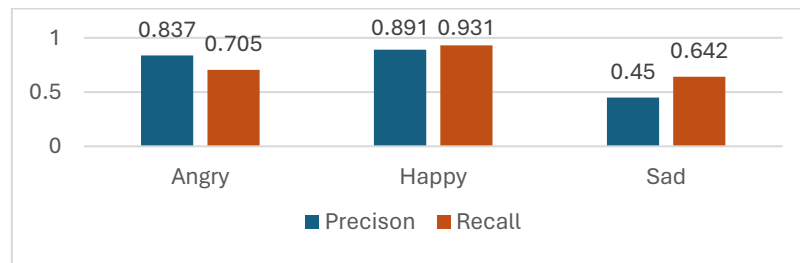
3.2 Selected Model

The emotion classification model that achieved the best results was a CNN trained on a dataset preprocessed with Gaussian blur using a kernel size of 5x5. This model surpassed all others in terms of accuracy, weighted average precision, and weighted average recall. Compared to alternative models trained on datasets with other preprocessing methods, this CNN exhibited superior performance across all three-evaluation metrics, highlighting the efficacy of Gaussian blur preprocessing in enhancing the model's ability to accurately classify emotions in images.

These results are obtained from the best-performing model trained on a Gaussian blur preprocessed dataset with a kernel size of 5x5:

Pre-processing	Accuracy	Weighted AVG Precision	Weighted AVG Recall
Gaussian Blur Kernel size (5x5)	0.788	0.788	0.789

Best Model - Gaussian Blur, Kernel Size (5x5) Performance By Classes:



This emotion classification model demonstrated varying performance across different emotion classes:

For the "happy" class, the model achieved the highest precision and recall scores compared to other classes. This indicates that a large proportion of images identified as happy were indeed happy (*Precision* = 0.891), and the model successfully identified most of the actual happy images (*Recall* = 0.931).

In contrast, the "angry" and "sad" classes exhibited lower precision and recall values, suggesting slight challenges in accurately distinguishing these emotions. The "sad" class particularly showed the weakest performance ($Recall = 0.642$, $Precision = 0.45$). The low recall value highlight significant challenges in the model's ability to accurately detect sadness. A precision of 0.45 suggests that approximately half of the model's classifications within the "sad" class were correct. These results possibly obtained due to similarities between facial expressions depicting sadness and anger, making the classification of these classes more complex for the model.

4 Conclusions and Discussion

In relation to our research question, "Can image processing techniques be utilized for image emotion classification, and are there preprocessing techniques that can enhance classification?" our project's findings shed light on the potential of image processing in addressing emotion classification tasks. We discovered that applying image processing techniques can enhance the models' performance compared to not using any preprocessing at all. Investigating various preprocessing methods is crucial in classification tasks, particularly those involving facial expressions, and fine-tuning their hyperparameters should be considered for achieving optimal results.

Our study revealed that low pass filters generally yielded the best results among the preprocessing techniques, confirming our initial hypothesis. These filters facilitate image smoothing, and enhance the visibility of facial expression features such as broad bright smiles or dark anger wrinkles while reducing noise, thus making the images easier to analyze. This smoothing effect found to be beneficial for facial emotions classification and therefore might be considered in similar tasks.

Furthermore, fine-tuning the preprocessing techniques hyperparameters could enhance the model's classification accuracy, as evidenced by our results. Different hyperparameter produced different outcomes, emphasizing that the classification performance depends not only on the chosen technique, but also on its specific parameters.

Our chosen model's low performance over the angry and sad emotions classes underscores the need for training the model over additional examples of these classes, to better identify the key features and patterns that distinguish these emotions. This might help to achieve better performances in classifying those emotions and better distinguish between these emotions.

5 Further Work

Emotion classification is an important task in computer vision and machine learning, and the applications of it expected to increase in the near future. Therefore, robust and well-preformed models are required for achieving the best results in this task over its different and varied applications in medical, healthcare, robot-

human interaction, security and other fields. This will require improving our existing models, that can be done with several techniques.

First, we suggest increasing the size of the dataset used for training, evaluating, and testing the model. Our dataset, which was manually collected from several online sources, is still relatively small for achieving a robust model, that effectively detects key features and patterns relevant to different facial emotions expressions and achieve accurate predictions of the emotion's classes.

Also, we suggest especially focusing on increasing the dataset with more examples of images expressing anger and sadness, given their lower precision and recall compared to the "happy" class. The number of classes should also be considered, adding other emotions, such as fear, surprise, excitement etc., so the model will be suited to a more diverse set of applications that require additional emotions classification.

In our project, the models were hyperparameters fine-tuned with a small number of options for each preprocessing techniques. Future work might consider a larger grid-search to find better parameters for these preprocessing techniques that might help the model to perform better. Also, other preprocessing techniques should be considered, especially other low-pass filters as we found they perform better compared to other examined preprocesses.

Although CNN are considered as preferred models for computer vision and image classification tasks, we might want to examine other models, such as random decision forest or other neural networks. In addition, we recommend considering different architecture for the CNN model, as we only examined one architecture. A different number of hidden layers, activation function or max pooling might lead to an improved model for this task, that will result in better performance.

To conclude, emotion classification is an important task these days. We see great value in performing further research in this field that will lead to advanced models that could be able to accurately capture human emotions expressions. Those models will lead to the development of well-performed applications in varying fields that potentially will improve daily life, existing workflows, and society overall in different aspects.

6 Bibliography

- [1] Maw, H. M., Lin, K. Z., & Mon, M. T. (2018, July). Preprocessing techniques for face and facial expression recognition. In *The 33rd International Technical Conference on Circuits/Systems, Computers and Communications (33rd ITC-CSCC)* (pp. 377-380).
- [2] Pitaloka, D. A., Wulandari, A., Basaruddin, T., & Liliana, D. Y. (2017). Enhancing CNN with preprocessing stage in automatic emotion recognition. *Procedia computer science*, 116, 523-529.
- [3] Kumar, A., Kaur, A., & Kumar, M. (2019). Face detection techniques: a review. *Artificial Intelligence Review*, 52, 927-948.
- [4] Cuimei, L., Zhiliang, Q., Nan, J., & Jianhua, W. (2017, October). Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. In *2017 13th IEEE international conference on electronic measurement & instruments (ICEMI)* (pp. 483-487). IEEE.
- [5] Pitaloka, D. A., Wulandari, A., Basaruddin, T., & Liliana, D. Y. (2017). Enhancing CNN with preprocessing stage in automatic emotion recognition. *Procedia computer science*, 116, 523-529.
- [6] Hemalatha, G., & Sumathi, C. P. (2016, February). Preprocessing techniques of facial image with Median and Gabor filters. In *2016 International Conference on Information Communication and Embedded Systems (ICICES)* (pp. 1-6). IEEE.
- [7] Chengeta, K., & Viriri, S. (2019). Image preprocessing techniques for facial expression recognition with canny and kirsch edge detectors. In *Computational Collective Intelligence: 11th International Conference, ICCCI 2019, Hendaye, France, September 4–6, 2019, Proceedings, Part II 11* (pp. 85-96). Springer International Publishing.
- [8] Chauhan, R., Ghanshala, K. K., & Joshi, R. C. (2018, December). Convolutional neural network (CNN) for image detection and recognition. In *2018 first international conference on secure cyber computing and communication (ICSCCC)* (pp. 278-282). IEEE.
- [9] Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)* (pp. 1-6). Ieee.
- [10] Lopes, A. T., De Aguiar, E., & Oliveira-Santos, T. (2015, August). A facial expression recognition system using convolutional networks. In *2015 28th SIBGRAPI conference on graphics, patterns and images* (pp. 273-280). IEEE.
- [11] Viola, P., & Jones, M. (2001, December). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001 (Vol. 1, pp. I-I)*. Ieee.

7 Appendices




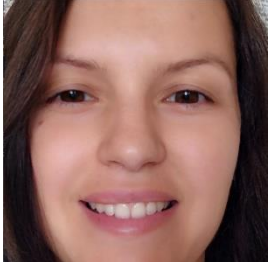
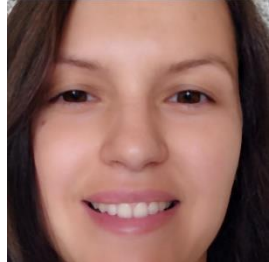
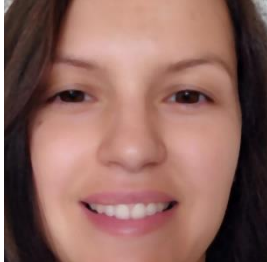

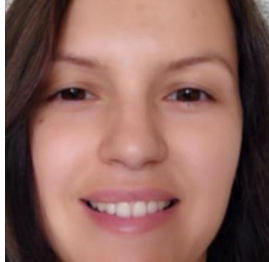
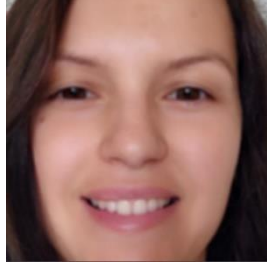



Appendix 1 - Data Access:






- <https://www.kaggle.com/datasets/sanidhyak/human-face-emotions/data>
- <https://www.kaggle.com/datasets/vfomenko/young-affectnet-hq/discussion>
- <https://www.kaggle.com/datasets/tapakah68/facial-emotion-recognition>
- <https://www.kaggle.com/datasets/mitech/stock2fer>

Appendix 2 - Face Detection



Appendix 3 - Preprocessing example - image enhancement

	Low Pass Filters		
	Kernel Size (3x3)	Kernel Size (5x5)	Kernel Size (7x7)
Gaussian Blur			
Median Blur			
Mean Blur			
	[[0,1,-1,0], [1,3,-3,-1], [1,3,-3,-1],[0,1,-1,0]]	[[0,1,-1,0], [1,5,-5,-1], [1,5,-5,-1],[0,1,-1,0]]	[[0,1,-1,0], [1,7,-7,-1], [1,7,-7,-1],[0,1,-1,0]]
2D Filter			

Without-pre-Processing	Contrast Enhancement	Thresholding	Canny	Merge Pre-processing
				

Appendix 4 – Models Evaluations

Pre-processing			Accuracy	Weighted AVG Precision	Weighted AVG Recall
Without Pre-processing			0.681	0.682	0.728
Low Pass Filters	Gaussian Blur	Kernel: 3x3	0.623	0.623	0.654
		Kernel: 5x5	0.788	0.788	0.789
		Kernel: 7x7	0.715	0.715	0.715
	Median Blur	Kernel: 3x3	0.779	0.779	0.779
		Kernel: 5x5	0.752	0.752	0.756
		Kernel: 7x7	0.660	0.660	0.667
	Mean Blur	Kernel: 3x3	0.633	0.633	0.667
		Kernel: 5x5	0.752	0.752	0.752
		Kernel: 7x7	0.678	0.678	0.675
	Filter 2D	Matrix 1 *	0.752	0.752	0.745
		Matrix 2 *	0.733	0.733	0.728
		Matrix 3 *	0.678	0.678	0.694
Contrast Enhancement			0.690	0.697	0.690
Thresholding			0.690	0.692	0.699
Canny			0.575	0.573	0.583
Merge Pre-process			0.575	0.575	0.572

*Matrix 1-[[0,1,-1,0], [1,3,-3,-1], [1,3,-3,-1],[0,1,-1,0]]

*Matrix 2- [[0,1,-1,0], [1,5,-5,-1], [1,5,-5,-1],[0,1,-1,0]]

*Matrix 3-[[0,1,-1,0], [1,7,-7,-1], [1,7,-7,-1],[0,1,-1,0]]