# A Project Report on Emotion Detection Using Image Processing Submitted to KIIT Deemed to be University In Partial Fulfilment of the Requirement for the Award of BACHELOR’S DEGREE IN COMPUTER SCIENCE & ENGINEERING BY **Debasish Ray : 21051647 Amrit Raj Bajapayi : 21051627** Under the Guidance of prof. Ramakant Parida

# School of Computer Engineering KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY BHUBANESWAR , ODISHA - 751024 APRIL , 2024

# KIIT Deemed to be University School of Computer Engineering Bhubaneswar , Odisha 751024 CERTIFICATE This is certify that the project entitled Emotion Detection Using Image Processing submitted by Debasish Ray : 21051647 Amrit Raj Bajapayi : 21051627

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2023-2024, under our guidance.

Date : 13/04/2024

Prof. Ramakant Parida

Project Guide

# Acknowledgements

# We are profoundly grateful to **Prof. Ramakant Parida** of Affiliation for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion.

Debasish Ray

Amrit Raj Bajapayi

# ABSTRACT

This project aims to detect users' emotions based on their facial expressions using Python (2.7), Open-CV, and NumPy. The system can analyze live feed from the camera or preexisting images to recognize human emotions. Emotion recognition in computer vision is a widely researched topic, and this project demonstrates the feasibility of predicting emotions using scanned images (testing data-set) compared to a training data-set. The objective is to develop a system capable of analyzing images and predicting the person's expression, which has been successfully achieved.  
  
Keywords : Face Recognition, Image Processing, Computer Vision, Emotion Detection, Open-CV

# Contents

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | Introduction | | 6 |
| 2 | Literature Review | | 7 |
| 3 | Problem Statement | | 7 |
| 4 | Introduction of OPEN-CV | | 8 |
| 5 | Implementation | | 9 |
| 5.1 | Importing Libraries | 9 |
| 5.2 | NumPy | 9 |
| 5.3 | Glob | 9 |
| 5.4 | Random | 9 |
| 6 | Code | | 10 |
| 6.1 | Code Explanation | 10 |
| 7 | DIFFERENT EMOTIONS THAT CAN BE DETECTED OUT OF AN IMAGE | | 12 |
|  |  | |  |
|  |  | |  |

# Chapter 1 Introduction

Image processing is a powerful method for improving images and extracting valuable information. It involves converting images into digital form and then manipulating them through various operations. Similar to signal processing, where signals are analyzed and manipulated, image processing deals with 2D images represented by pixel values ranging from 0 to 255. This allows for a wide range of operations to be performed on images, including enhancement, filtering, and feature extraction, enabling valuable insights to be gleaned from visual data.

The method begins by converting an image into a 2D matrix. This process involves three primary steps:

1. **Scanning the image:** In this step, a raw image is obtained for processing. The image is represented as pixels, capturing its visual information. The goal here is to extract data suitable for computational analysis.

2. **Processing and enhancing the image:** Next, the image is digitized using a digitizer, which samples and quantizes the input signals. A high sampling rate is desirable for good resolution, while a high quantization level allows for better perception of different shades using grayscale.

1. **Analyzing the obtained result:** The processed image reveals various properties of the original image and may involve further classification based on these properties.

Converting a color image to grayscale can be achieved through two main methods:

A) **Average Method:** This approach calculates the mean of the red, green, and blue color channels in the image. The formula used is: Grayscale = (R + G + B) / 3. However, this method may sometimes result in a black image due to an equal contribution from each color channel, resulting in a lack of contrast.

B) **Weighted or Luminosity Method:** To address the limitations of the Average Method, the Weighted or Luminosity Method is employed. In this method, the contribution of each color channel is adjusted based on its perceived brightness. Typically, the presence of red is reduced, while green's contribution is increased, with blue falling between these two. The equation for this method is: Grayscale = (0.3 \* R) + (0.59 \* G) + (0.11 \* B). These weights are chosen based on the wavelength patterns of the colors, with blue having the shortest wavelength and red the longest.

This method ensures that the resulting grayscale image maintains a balanced representation of the original color image while taking into account the human perception of brightness associated with different color channels.

# Chapter 2 Literature Review

1. Mr. Ashwini, Mr. Jacob, and Dr. Jubilant from St. Joseph’s College of Engineering conducted an analysis of an emotion recognition system, identifying various strengths and weaknesses.

2. In their paper titled "Emotion Recognition from Facial Expressions and Its Control Using Fuzzy Logic," Mr. Aruna Chakraborty, Mr. Amit Konar, Mr. Uday Kumar Chakraborty, and Mr. Amita Chatterjee discussed a fuzzy logic-based approach to emotion recognition, achieving an accuracy of approximately 90%.

1. Carlos Busso, Zhigang Deng, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann, and Shrikanth Narayanan conducted an analysis of facial expression classifiers and acoustic emotion classifiers, examining their respective strengths and weaknesses.

# Chapter 3 Problem Statement :

This project aims to create an emotion detection system leveraging image processing methods. Emotion detection holds significant importance across several domains, including human-computer interaction, affective computing, and mental health monitoring. The primary objective is to develop an efficient algorithm capable of accurately recognizing and categorizing human emotions based on facial expressions captured in images. The focus lies on designing a robust solution that can analyze facial features, extract pertinent information, and classify emotions into predefined categories like happiness, sadness, anger, fear, disgust, and surprise. The system must be adaptable to handle variations in lighting conditions, facial orientations, and expressions to ensure dependable performance in real-world settings. Additionally, the project will explore various techniques for data preprocessing, feature extraction, and classification, utilizing machine learning or deep learning algorithms to achieve superior accuracy and scalability. Ultimately, the proposed emotion detection system aims to advance research in affective computing and facilitate the seamless integration of emotion-aware technologies across diverse domains.



Fig :- Data Recording System

# Chapter 4 Introduction to OPEN-CV

Open-CV, or Open Computer Vision Library , is a comprehensive and freely available library renowned for its extensive collection of over 2500 algorithms, tailor-made for tasks in computer vision and machine learning projects. These algorithms serve a multitude of purposes, including face recognition, object identification, camera movement tracking, and scenery recognition. Open-CV boasts a vibrant community of approximately 47,000 active contributors, underscoring its widespread adoption across various sectors, both private and public.

A recent addition to Open-CV's arsenal is GPU Acceleration , augmenting its existing functionalities. While this feature is not yet fully matured, it effectively handles a broad range of operations. GPU Acceleration harnesses CUDA to leverage the computational capabilities of GPUs, benefiting from libraries like NPP (NVIDIA Performance Primitives). Its appeal lies in its accessibility, enabling users to tap into GPU processing power without requiring advanced knowledge of GPU programming. Notably, within the GPU Module, direct modification of image features is not feasible; instead, users must first copy the original image before applying edits.

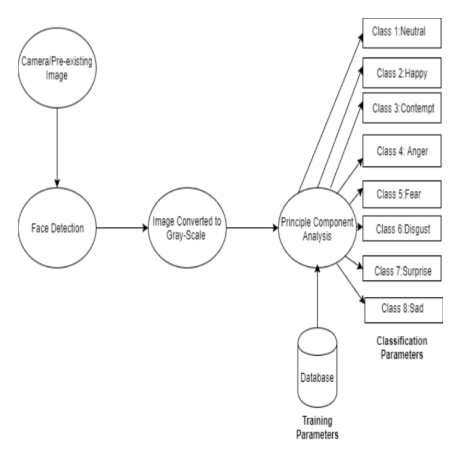


Fig :- Flowchart for emotion algorithm

# Chapter 5 Implementation

Let's start by taking an image in either .jpg or .png format and use image processing techniques to detect emotions from the subject in the image. Here, 'subject' refers to any living entity from which emotions can be identified.

A. Importing Libraries

For the effective execution of this project, you will need to install certain Python packages. Ensure you have Python 2.7.x, NumPy, Glob, and Random installed on your system. Once Python is set up, typically on the C drive, launch Python IDLE, import these packages, and begin your coding.

B. NumPy

- NumPy is a fundamental library for Python, widely used in scientific computing. It supports the use of multidimensional arrays, offering a variety of mathematical operations to manipulate these arrays.

- An array's dimension is referred to as its axis.

- The number of axes an array has is described as its rank. For example, the array `A = [1, 2, 3, 4, 5]` consists of five elements arranged in one dimension, thus it has a rank of 1.

- Consider another example: `B = [[1, 2, 3, 4], [5, 6, 7, 8]]`. This is a two-dimensional array where the first dimension contains two elements and the second dimension four, making its rank 2.

C. Glob

- The Glob module in Python is used to find all the path names matching a specified pattern according to the rules used by Unix shell. Glob returns the full path names that match the given pattern.

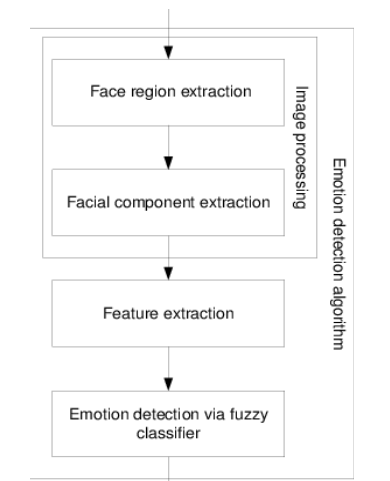
D. Random

The Random module in Python is utilized for generating random numbers or selecting random items from a list. This module includes functions that facilitate operations involving randomness.

Key Functions in the Random Module:

1) `randint(m, n)`: This function generates a random integer `x` where `m <= x <= n`.

2) `randrange(dog, cat, mouse, lion)`: Incorrectly used in the context, but typically `randrange` provides a randomly selected element from a range. For lists, the equivalent function is `choice`, which would be used like `choice([dog, cat, mouse, lion])` to select a random element from the specified list.

Fig :- Flowchart for Emotion Detection

# Chapter 6 CODE

from keras.models import load\_model

from time import sleep

from keras.preprocessing.image import img\_to\_array

from keras.preprocessing import image

import cv2

import numpy as np

face\_classifier = cv2.CascadeClassifier(r'C:\Users\KIIT\OneDrive\Desktop\Emotion Detection\Emotion\_Detection\_CNN\haarcascade\_frontalface\_default.xml')

classifier =load\_model(r'C:\Users\KIIT\OneDrive\Desktop\Emotion Detection\Emotion\_Detection\_CNN\model.h5')

emotion\_labels = ['Angry','Disgust','Fear','Happy','Neutral', 'Sad', 'Surprise']

cap = cv2.VideoCapture(0)

while True:

    \_, frame = cap.read()

    labels = []

    gray = cv2.cvtColor(frame,cv2.COLOR\_BGR2GRAY)

    faces = face\_classifier.detectMultiScale(gray)

    for (x,y,w,h) in faces:

        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,255),2)

        roi\_gray = gray[y:y+h,x:x+w]

        roi\_gray = cv2.resize(roi\_gray,(48,48),interpolation=cv2.INTER\_AREA)

        if np.sum([roi\_gray])!=0:

            roi = roi\_gray.astype('float')/255.0

            roi = img\_to\_array(roi)

            roi = np.expand\_dims(roi,axis=0)

            prediction = classifier.predict(roi)[0]

            label=emotion\_labels[prediction.argmax()]

            label\_position = (x,y)

            cv2.putText(frame,label,label\_position,cv2.FONT\_HERSHEY\_SIMPLEX,1,(0,255,0),2)

        else:

            cv2.putText(frame,'No Faces',(30,80),cv2.FONT\_HERSHEY\_SIMPLEX,1,(0,255,0),2)

    cv2.imshow('Emotion Detector',frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):

        break

cap.release()

cv2.destroyAllWindows()

**Code Explanation :-**

The code you've provided is a Python script using Open-CV and Keras libraries to create a real-time emotion detection system from webcam video input. It utilizes a pre-trained convolutional neural network (CNN) model for classifying emotions based on facial expressions. Here’s a breakdown of what each part of the script does:

1. Import Libraries:

- `load\_model` from Keras: Loads the pre-trained CNN model.

- `sleep` from time: Not used in the script.

- `img\_to\_array` and `image` from Keras: Convert images to arrays suitable for model input.

- `cv2` for Open-CV: Handles all operations related to image processing and capturing video input.

- `np` for NumPy: Provides support for large, multi-dimensional arrays and matrices, along with a large collection of mathematical functions to operate on these arrays.

2. Load Pre-trained Models:

- `face\_classifier`: An Open-CV Haar cascade classifier, pre-trained for detecting frontal faces (`haarcascade\_frontalface\_default.xml`).

- `classifier`: A pre-trained CNN model loaded from a `.h5` file designed to classify emotions.

3. Setup Emotion Labels:

- `emotion\_labels`: A list that maps the output of the CNN (an index) to an emotion label.

4. Initialize Webcam:

- `cap`: Captures video through the first connected webcam (`cv2.VideoCapture(0)`).

5. Main Loop:

- Captures frames from the webcam in a loop.

- Converts each frame to grayscale—necessary for the face detection step.

- Detects faces in the grayscale image using the Haar cascade classifier.

- For each detected face:

- Draws a rectangle around the face.

- Extracts the region of interest (ROI) which is the face, resizes it to 48x48 pixels (the input size expected by the CNN).

- Normalizes the pixel values of ROI by dividing by 255.0 (to scale between 0 and 1).

- Converts the ROI to a 4D array (batch size 1, height, width, channels) required by Keras for prediction.

- Predicts the emotion using the CNN and finds the label with the highest probability.

- Displays the predicted emotion label on the frame near the corresponding face.

6. Output Display and Cleanup:

- Shows the frame with detected faces and their predicted emotions in a window titled 'Emotion Detector'.

- Breaks out of the loop if the 'q' key is pressed.

- Releases the webcam and destroys all Open-CV windows to free resources.

This script effectively applies computer vision and machine learning techniques to detect and display emotions in real-time using facial expressions captured via a webcam. It's a practical example of how to integrate deep learning models and image processing for real-world applications.

# Chapter 7 DIFFERENT EMOTIONS THAT CAN BE DETECTED OUT OF AN IMAGE:

# Neutral

# Happy

# Anger

# Disgust

# Surprise

# Fear

# Sad

1. Contempt

|  |  |  |
| --- | --- | --- |
| Screenshot 2024-04-13 011422 | Screenshot 2024-04-13 011519 | Screenshot 2024-04-13 011612 |

Fig :- Output

# Chapter 8 STEPS INVOLVED TO PERFORM EMOTION DETECTION USING OPEN-CV PYTHON

1) Begin by assembling a dataset for training the emotion detection system. You can create a custom dataset by gathering and labeling a diverse collection of images, ensuring a rich dataset for accurate analysis. Alternatively, utilizing an established database can save time and provide a robust foundation for training.

2) Organize the dataset into two main directories: one for storing all image files and another for maintaining detailed information about the various emotion categories represented in the images.

3) Process the images using the specified Python script. Store the resulting output images in a separate directory, categorizing them by the type of emotion detected and their corresponding labels.

4) For emotion recognition, we will predominantly utilize the Fisher Face classifier in OpenCV, known for its effectiveness in distinguishing facial features relevant to different emotions.

5) To maximize face detection accuracy, employ a sequence of the four predefined classifiers provided by OpenCV. This approach helps in capturing as many facial details as possible from the images.

6) Divide the dataset into a training set and a test set. The training set will be used to train the model on recognizing different emotions by analyzing a substantial number of images. The test set will assess the accuracy of the trained classifier.

7) Ensure all images are standardized in terms of size and properties to maintain consistency during training and testing, which enhances model reliability.

8) Each image in the dataset should be processed by converting it to grayscale, cropping to focus on the subject, and saving it in an organized directory for systematic training and testing.

9) Construct the training dataset with 80% of the images, using them to train the emotion recognition model. Use the remaining 20% for testing the model’s classification accuracy. Iterate this process to refine and enhance the model’s performance.

This database is a valuable resource for emotion detection, encompassing a wide range of emotional expressions. It is currently available in two editions, with a third under development. The initial version, known as "CK," begins by identifying a neutral emotional state before progressing to more intense emotional expressions. The subsequent version, "CK+," has enhanced the processing speed by approximately 25% and expanded the variety of subjects by about 30%. The forthcoming third version aims to combine the features of both existing versions, such as including additional functionality like a 30-degree rotation from the frontal view to provide more comprehensive facial analysis capabilities.

# Chapter 9 Application & Scope in Future

Computer Vision is a rapidly evolving field with extensive ongoing research. Within this domain, emotion detection stands out as a critical area due to its vast potential applications. Emotion detection facilitates various tasks and processes, revealing the expansive possibilities within this specialization.

Applications of emotion detection include:

A. App and Product Development: Emotion recognition significantly enhances software engineering by optimizing user experience testing. The comfort and emotional response of users to software products are crucial for their market success. Understanding the emotional impacts of product design and functionality is vital for developers and designers in the industry.

B. Enhanced Learning Environments: Research shows that emotional states can either enhance or inhibit learning. While positive emotions might seem counterproductive in learning scenarios, slightly negative emotions may encourage analytical thinking and effectiveness in performing critical tasks.

C. Advanced Web Development: As the internet grows, the need for sophisticated user data analysis becomes more pronounced. Emotion detection can refine user profiling, leading to more targeted content and advertising, enhancing user engagement and satisfaction.

D. Immersive Gaming Experiences: The gaming industry can greatly benefit from incorporating emotion detection. By understanding common emotional responses, game developers can create more engaging and emotionally resonant gaming experiences.

# Chapter 10 Challenges:

One of the primary challenges in emotion detection is accurately distinguishing between similar emotions, such as distinguishing "Disgust" from "Sadness" or "Surprise" from "Happy." In our project, using the Cohn-Kanade (CK and CK+) Database, we achieved about 70% accuracy. Improvements could be made by expanding the dataset, particularly by increasing the number of images for underrepresented emotions such as "sad," "fear," and "contempt," or by refining the classification model to enhance precision.

# Chapter 11 Conclusion

Artificial Intelligence (AI) is adept at tackling complex challenges like emotion detection. However, the complexity of this task increases substantially with the scale of the image data-set involved. As the number of images grows, distinguishing subtle emotional cues becomes more intricate, requiring more sophisticated algorithms and computational resources. This complexity highlights the need for advanced AI techniques and improved data processing methods to efficiently handle large data-sets while maintaining high accuracy in emotion recognition.  
Artificial Intelligence plays a crucial role in addressing complex challenges like emotion detection. This task becomes increasingly complicated as the size of the image dataset grows. Like humans who occasionally misinterpret emotions, our AI systems also face challenges in accurate recognition. Achieving an accuracy rate of approximately 83% demonstrates the capabilities and limitations of our current technology. In our recent project, we utilized Python 2.7, OpenCV, and the Cohn-Kanade (CK and CK+) databases to identify various human emotions, providing valuable insights into the nuances of emotional recognition.

# References

[1] V Gent, P. (2016). Emotion Recognition with Python, OpenCV and a

Face Dataset. A tech blog about fun things with Python and embedded

electronics. Retrieved from:

http://www.paulvangent.com/2016/04/01/emotion-recognition-withpython-opencv-and-a-face-dataset/

[2] Engineers Garage. (n.d.). Image Processing - Tutorial & Applications. Retrieved from: https://www.engineersgarage.com/articles/image-processing-tutorialapplications

[3] Python Software Foundation. (n.d.). glob — Unix style pathname pattern expansion. Retrieved from: https://docs.python.org/2/library/glob.html

[4] OpenCV. (n.d.). OpenCV. Retrieved from: https://opencv.org/

[5] The Linux Documentation Project. (n.d.). Globbing with Linux. Retrieved from: http://www.tldp.org/LDP/GNU-Linux-Tools-Summary/html/x11655.htm

[6] Kanade, T., Cohn, J. F., & Tian, Y. (2000). Comprehensive database for facial expression analysis. Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), Grenoble, France, 46-53.

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010), San Francisco, USA, 94-101

[7] Python Software Foundation. (n.d.). random — Generate pseudo-random numbers. Retrieved from: http://docs.python.org/3.4/library/random.html

[8] Tutorialspoint. (n.d.). Digital Image Processing. Retrieved from: https://www.tutorialspoint.com/dip/

[9] Psych Mnemonics. (2015). Primary Emotions. Retrieved from: https://psychmnemonics.wordpress.com/2015/07/03/primary-emtions

[10] SciPy. (n.d.). NumPy Quickstart Tutorial. Retrieved from: https://docs.scipy.org/doc/numpy-dev/user/quickstart.html

[11] Kanade, Cohn, & Tian (2000) and Lucey et al. (2010) in any paper of mine or my collaborators that makes any use of the database. The references are:

Kanade, T., Cohn, J. F., & Tian, Y. (2000). Comprehensive database for facial expression analysis. Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), Grenoble, France, 46-53.

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010), San Francisco, USA, 94-101.

[12] Pulli, K., Baksheev, A., Kornyakov, K., & Eruhimov, V. (2012). Realtime Computer Vision with OpenCV. Queue, 10(4), 40-56. DOI: http://dx.doi.org/10.1145/2181796.2206309

[13] Kim, Y. H., & Park, J. B. (2015). Emotion Detection Algorithm Using Frontal Face Image

[14] Verghese, A. A., Cherian, J. P., & Kizhakkethottam, J. J. (2015). Overview on Emotion Recognition System. INSPEC Accession Number: 15508917.

[15] Chakraborty, A., Konar, A., Chakraborty, U. K., & Chatterjee, A. (2009). Emotion Recognition From Facial Expressions and Its Control Using Fuzzy Logic. INSPEC Accession Number: 10705009

[16] Busso, C., Deng, Z., Yildirim, S., Bulut, M., Lee, C. M., Kazemzadeh, A., Lee, S., Neumann, U., & Narayanan, S. (2004). Analysis of Emotion Recognition using Facial Expressions, Speech and Multimodal Information

**EMOTION DETECTION USING OPEN-CV**

DEBASISH RAY

21051647

**Abstract :** Our project aims to develop a Emotion Detection System that accurately and efficiently recognizes if the Person is happy or sad or surprise or anything else . To achieve this, we explore different methods for detecting the emotion on the face , including physiological measures such as eye movements . We also investigate the use of Open-CV to improve the accuracy of our system .

**Individual Contribution and Findings :** As part of my contribution to the project, I played a crucial role in developing the facial detection code. The facial detection code was an essential component of the project, and to accomplish this, I utilized the dlib library, which is a powerful open-source tool for deep learning and computer vision. The dlib library enabled me to capture facial images and detect facial landmarks with a high degree of accuracy and efficiency, which were both critical to the success of the project. Through my efforts, I was able to make a significant contribution to the project and help achieve our goals with confidence.

**Individual contribution to project report preparation :** I would like to bring to your attention my contribution to the group project report. Specifically, I was responsible for drafting the sections on system design and literature review. In the system design section, I provided a comprehensive evaluation of the model design, taking into account all relevant factors. This enabled me to provide an in-depth analysis of the relevant literature, and determine the best practices to be applied in the report. Overall, my contribution to the group project report was instrumental in ensuring its quality and academic rigor.

**Individual contribution for project presentation and demonstration :** During the project presentation and demonstration, I was responsible for the design and implementation of the model. To do so, I prepared a comprehensive presentation that covered the model's functionality, showcasing its working, and outlining potential areas for further research and development. Overall, the presentation was received positively, and it helped to generate valuable discussions and suggestions for future improvements.

**Full Signature of Supervisor Full Signature of the Student**

**….…………………………… ………………………………..**

**EMOTION DETECTION USING OPEN-CV**

AMRIT RAJ BAJAPAYI

21051627

**Abstract :** Our project aims to develop a Emotion Detection System that accurately and efficiently recognizes if the Person is happy or sad or surprise or anything else . To achieve this, we explore different methods for detecting the emotion on the face , including physiological measures such as eye movements . We also investigate the use of Open-CV to improve the accuracy of our system .

**Individual Contribution and Findings :** As part of my contribution to the project, I played a crucial role in developing the facial detection code. The facial detection code was an essential component of the project, and to accomplish this, I utilized the dlib library, which is a powerful open-source tool for deep learning and computer vision. The dlib library enabled me to capture facial images and detect facial landmarks with a high degree of accuracy and efficiency, which were both critical to the success of the project. Through my efforts, I was able to make a significant contribution to the project and help achieve our goals with confidence.

**Individual contribution to project report preparation :** I would like to bring to your attention my contribution to the group project report. Specifically, I was responsible for drafting the sections on system design and literature review. In the system design section, I provided a comprehensive evaluation of the model design, taking into account all relevant factors. This enabled me to provide an in-depth analysis of the relevant literature, and determine the best practices to be applied in the report. Overall, my contribution to the group project report was instrumental in ensuring its quality and academic rigor.

**Individual contribution for project presentation and demonstration :** During the project presentation and demonstration, I was responsible for the design and implementation of the model. To do so, I prepared a comprehensive presentation that covered the model's functionality, showcasing its working, and outlining potential areas for further research and development. Overall, the presentation was received positively, and it helped to generate valuable discussions and suggestions for future improvements.

**Full Signature of Supervisor Full Signature of the Student**

**….…………………………… ………………………………..**