**DETECT FACES AND ANALYSE EMOTIONS USING FACIAL EMOTION RECOGNITION API WITH IBM CLOUD**

**INDEX:**

1. Introduction

a. Overview

b. Purpose

2. Literature Survey

a. Existing problem

b. Proposed solution

3. Theoretical Analysis

a. Block diagram

b. Hardware / Software designing

4. Experimental Investigations

5. Flowchart

6. Result

7. Advantages & Disadvantages

8. Applications

9. Conclusion

10. Future Scope

11. Bibliography

12. Appendix

a. Source code

b. UI output Screenshot

**1.INTRODUCTION**:

**a) Overview:** We know that emotions play a major role in a Human life. At different kind of moments or time Human face reflects that how he/she feels or in which mood he/she is. Humans are capable of producing thousands of facial actions during communication that vary in complexity, intensity, and meaning. Emotion or intention is often communicated by subtle changes in one or several discrete features. The addition or absence of one or more facial actions may alter its interpretation. In addition, some facial expressions may have a similar gross morphology but indicate varied meaning for different expression intensities. In order to capture the subtlety of facial expression in non-verbal communication, I will use an existing simulator which will be able to capture human emotions by reading or comparing facial expressions. This algorithm automatically extracts features and their motion information, discriminate subtly different facial expressions, and estimate expression intensity.Fig1 is showing how Emotion recognition using brain activity performs its task.

**b) Purpose:** The main aim of this project is to detect face and should analyse the emotion of that face.This project will be very useful in the real time environment.By using Api’s we can detect the emotions.Human emotion recognition plays an important role in the interpersonal relationship. ... Emotions are reflected from speech, hand and gestures of the body and through facial expressions. Hence extracting and understanding of emotion has a high importance of the interaction between human and machine communication.

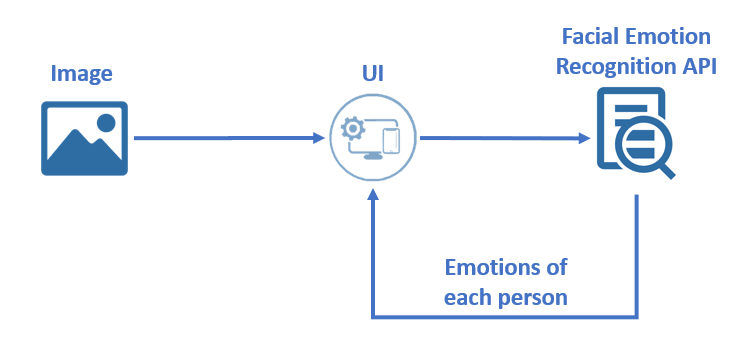
**2.LITERATION SURVEY:**

**a) Existing problem: Sometimes we all are intrested to identify the emotion of a person. But we are having less resources to extract the emotion of a person. To resolve this issue we are introducing our project. By using our project we can simply recognize the emotion of a person.**

**b)Proposed System: To resolve the problem which we are facing to recognize/identify** theemotion of a person we are introducing our project. First user should copy the link of that particular persons face and copy it and user should paste the link in our proposed project application. If we enter the correct link then the we can get the correct output nothing but the emotion of a person. If there is any error in the code we will not get the correct emotion/ sometimes we may not get any output.

**3.THEOTRICAL ANALYSIS:**

**a)Block diagram:**

****

**b) Hardware/Software designing**

**Software Requirements:**

* OS-Windows XP,7,8,10
* Spyder Software
* Anaconda Command Prompt
* Flask framework
* Browser

**Hardware Components**

* Processor – i3
* Hard Disk Storage – 5GB
* RAM – 1GB

**Required packages:**

1. Open anaconda prompt.
2. Type "pip install requests” and click enter.
3. Type “pip install Flask” and click enter.

**4.EXPERIMENTAL INVESTIGATIONS:**

* **Flask:** Web framework used for building Web applications.
* **API:** An application programming interface (API) is a computing interface which defines interactions between multiple software intermediaries.

**5.FLOW CHART:**

Find below the project flow to be followed while developing the project.

* Signup for Rapid API.
* Subscribe to the API.
* Test the API with static images.
* Build a Web application using flask that integrates with the API.

**6.RESULT**: By using this application finally user can get the emotion of a person.

**7.ADVANTAGES AND DISADVANTAGES:**

**Advantages:**

* Automatically detects the emotion.
* Face detection can also be used to auto focus cameras or to count how many people have entered an area.

**Disadvantages:**

* Blur images cannot recognize the emotion correctly.

**8. Applications:**

* Detection of mental disorders
* Synthetic human expressions

**9.CONCLUSION:**

* By doing this project which is detect faces and andanalyse emotion recognition API with IBM cloud, we have learnt how to use APIs, how to hit the APIs, send and retrieve the data and how to build web applications using flask framework. We also learnt how API works.

**10.FUTURE SCOPE:**

**I**n future we can include more features to our applications, so it will useful to our project.

**11.BIBILOGRAPHY:**

https://www.marketsandmarkets.com/ResearchInsight/

emotion-detection-recognition-market.asp.Accessed: 2018-06-20.

[2]nviso helping ﬁnancial advisors understand their clients’ true ﬁnancial

needs with emotional intelligence.https://www.ibm.com/case-studies/

t338899k54153u05.Accessed: 2018-06-20.

[3]Verilooksdk: Face identiﬁcation for stand-alone or web applications.

http://www.neurotechnology.com/verilook.html.Accessed: 2018-06-20.

[4] AZCARATE, A., HAGE LO H, F., VAN DE SAN DE , K. , AND VALE NT I, R .

Automatic facial emotion recognition.

[5] BAZ ZO , J. , AN D LAMAR, M. Recognizing facial actions using gabor

wavelets with neutral face average difference., 01 2004.

[6] BERNIN, A., M ¨

ULLER, L., GHOSE, S., VON LUCK, K. , GRECOS, C .,

WANG, Q ., A ND VOGT, F. Towards more robust automatic facial

expression recognition in smart environments. Proc. of the 10th Int.

Conf. on Perv. Tech. Related to Assistive Environments (2017), 37–44.

[7] COUNCIL, N. R., ET A L. The polygraph and lie detection. committee to

review the scientiﬁc evidence on the polygraph. division of behavioral

and social sciences and education. Washington, DC: The National

Academic Press. Retrieved 7, 7 (2003), 09

* Bassili J. N. (1979). Emotion recognition: The role of facial movement and the relative importance of upper and lower areas of the face. Journal of Personality and Social Psychology, 37, 2049–2058. [[CrossRef]](https://dx.doi.org/10.1037/0022-3514.37.11.2049) [[PubMed]](https://www.ncbi.nlm.nih.gov/pubmed/521902)
* Berry D. S. (1990). What can a moving face tell us? Journal of Personality and Social Psychology, 58, 1004–1014. [[CrossRef]](https://dx.doi.org/10.1037/0022-3514.58.6.1004) [[PubMed]](https://www.ncbi.nlm.nih.gov/pubmed/2391636)
* Blair R. J. R. Colledge E. Murray L. Mitchell D. G. V. (2001). A selective impairment in the processing of sad and fearful expressions in children with psychopathic tendencies. Journal of Abnormal Child Psychology, 29, 491–498. [[CrossRef]](https://dx.doi.org/10.1023/A:1012225108281) [[PubMed]](https://www.ncbi.nlm.nih.gov/pubmed/11761283)

**12. SOURCE CODE(App.py):**

#importing required libraries

from flask import Flask, request, render\_template

import numpy as np

import re

import requests

import pandas as pd

#initializing the flask app

app = Flask(\_\_name\_\_)

def check(output):

url = "https://facial-emotion-recognition.p.rapidapi.com/cloudVision/facialEmotionRecognition"

querystring = {"source":output,"sourceType":"url"}

payload = '''{\r\n \"source\": "'''+output+'''" ,\r\n \"sourceType\": \"url\"\r\n}'''

headers = {

'content-type': "application/json",

'x-rapidapi-key': "d730d682demshbfaf2fc584babdap15184djsnd2eec3eec4be",

'x-rapidapi-host': "facial-emotion-recognition.p.rapidapi.com"

}

response = requests.request("POST", url, data=payload, headers=headers, params=querystring)

print(response.text)

a = response.json()['emotions']

#converting the output to a dataframe using pandas

df = pd.DataFrame(data=a)

b = [i for i in range(1,len(a)+1)]

ind={}

for i in b:

ind[i-1]="Person {:d}".format(i)

#renaming the index and column names

df=df.rename(index=ind)

df.columns = ['Joy','Sorrow','Anger','Surprise','UnderExposed','Blurred','Hear Wear']

df=df.replace(['VERY\_UNLIKELY','UNLIKELY','VERY\_LIKELY','LIKELY'], ["Very Unlikely","Unlikely","VeryLikely","Likely"])

#converting the dataframe to HTML table

df\_html = df.to\_html(classes='mystyle', justify="center",col\_space=90)

return df\_html

#home page

@app.route('/')

def home():

return render\_template('base.html')

#Summarizer page

@app.route('/predict',methods=['POST'])

def predict():

output=request.form['output']

df\_html=check(output)

return render\_template('base.html',table\_html=df\_html)

if \_\_name\_\_ == "\_\_main\_\_":

app.run(debug=True)

**Source Code(Emotion.py):**

import requests

url = "https://facial-emotion-recognition.p.rapidapi.com/cloudVision/facialEmotionRecognition"

querystring = {"source":"https://images.unsplash.com/photo-1527631120902-378417754324?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&auto=format&fit=crop&w=2250&q=80","sourceType":"url"}

payload = "{\r\n \"source\": \"https://images.unsplash.com/photo-1527631120902-378417754324?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&auto=format&fit=crop&w=2250&q=80\",\r\n \"sourceType\": \"url\"\r\n}"

headers = {

'content-type': "application/json",

'x-rapidapi-key': "d730d682demshbfaf2fc584babdap15184djsnd2eec3eec4be",

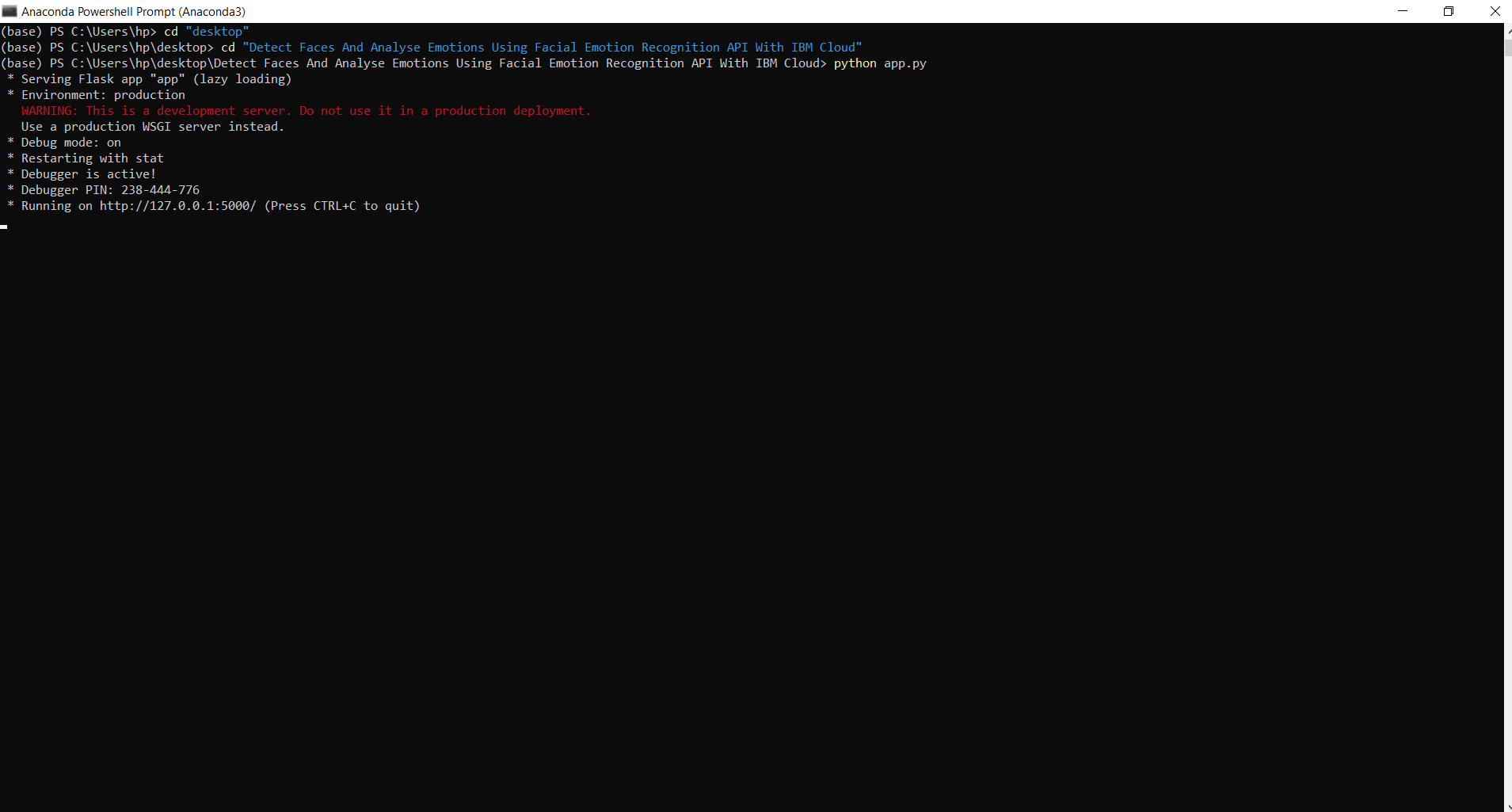
'x-rapidapi-host': "facial-emotion-recognition.p.rapidapi.com"

}

response = requests.request("POST", url, data=payload, headers=headers, params=querystring)

print(response.json())

**UI Output Screenshot:**

****

