**Facial Emotion Recognitio**

Under the guidance of Mr. Jitendra and Mr. Yogesh, I (Ishant Dahiya) am making a face and emotion recognition model to predict if the user is happy, sad, neutral, angry or in fear.

**Data Collection**

Databases used:

* <http://download.visgraf.impa.br/>
* <http://www.consortium.ri.cmu.edu/data/ck/>
* <http://grail.cs.washington.edu/projects/deepexpr/FERG_DB_256.zip>

Also to collect more images, I made a scraper using Selenium and urlib.request. I used these libraries becuase [https://shutterstock.com](https://shutterstock.com/) used dynamically loaded pages with javascript, so I used firefox header with Selenium to get the data.

**Data Arrangement**

The data was spread across different folders, coded differently in all the databases.

I used some simple python scripts, to gather them into labelled folder so the model could read them easily.

#!/usr/bin/env python3

# -\*- coding: utf-8 -\*-

"""

Created on Wed Jan 15 22:49:50 2020

@author: ishant

"""

import urllib.request

import random

from selenium import webdriver

import os

from datetime import datetime

start = datetime.now()

def scraper(url):

global count

driver = webdriver.Firefox(executable\_path = "/home/ishant/ishant\_linux/facial-emotion-recognition/geckodriver")

driver.get(url)

results = driver.find\_elements\_by\_class\_name("z\_g\_a")

for result in results:

if result.get\_attribute('src') == None:

print("the error is here")

continue

img\_name = "neutral" + str(random.randrange(10,100000)) + ".jpg"

fullfilename = os.path.join("data/scraper/neutral", img\_name)

urllib.request.urlretrieve(result.get\_attribute('src'), fullfilename)

# print(fullfilename + " Successfully Saved!")

count =+ 1

driver.quit()

print("Scraper function successfully executed")

urlpage = [None] \* 5

for i in range (0, 5):

if i==0:

urlpage[i] = "https://www.shutterstock.com/search/neutral+facial+Expression"

else:

urlpage[i] = "https://www.shutterstock.com/search/neutral+facial+Expression?page=" + str(i+1)

print(urlpage)

count = 0

# print(urlpage)

for x in urlpage:

scraper(x)

print("Scraper function successfully called and executed")

end = datetime.now()

total\_time = end - start

print("Total number of files saved :: " + str(count) + " in Time :: " + str(total\_time) )