



Study on Desktop Partner Based on Python and Machine Learning

Abhishek Verma¹, Prabhat Chandra Gupta²

1. BTECH (CSE) 8th sem. GU, GREATER NOIDA

2. Assistant Professor, GU, GREATER NOIDA

Abstract- Emotion detection plays a vital role in recent days. Through this paper, we tend to gift a approach for building this desktop application using facial expressions. Our application will acknowledge the mood of a user through camera. For an instance, if a user is unhappy, then system will mechanically show some message along with user's mood on the screen. For this whole system, we tend to square measure victimisation technologies like Machine Learning, Python, Data Mining etc. The testing dataset is being compared to coaching dataset and so motion is expected. the target of this paper is to develop a system which may analyze the image and predict the expression of an individual. Its some major Keywords: Image process, Python, Machine Learning.

1. INTRODUCTION

Detection of mood plays a major role in human dialogue. The Desktop Partner is made victimisation an Machine Learning algorithmic program. The partner offers you some message as a real person. In projected system, we are going to see a replacement approach for building this application victimisation gestures. Our desktop partner acknowledges the mood of a user through camera.

2. DESKTOP PARTNER: DEFINATION

Desktop partner offers you some pleasant replies when you are tired or bored. The desktop partner mechanically detects our mood through camera and gives us entertaining replies along with our mood on our screen. The DP is made with an ML process. It chats with you as an actual person which doesn't feel the user he is getting replies from a bot.

3. EXIXTING WORK

Mood analysis can be done by victimisation facial gestures or by body pose. In projected system we tend to use facial gestures for sentiment analysis. Biometric identification could be a method of recognizing a person's face through technology. A Biometric identification system uses biometric to map facial expression from a photograph or video. It compares the knowledge with a info of better-known faces to search out a match. Each and every system of sentiment recognition needs a dataset.

Dataset is the most important portion for the judgment of the facial portion. For dataset preparation features are extracted from face portion and after that features are stored in database, Database is used for detecting the emotions by applying different algorithms. Features of image are extracted through preprocessing, Feature Extraction and so

classification.

4. TECHNOLOGIES TO BE USED

4.1 Image Processing:

So as to extract some info and so as to induce increased image, Image Processing will be used. It is a systematic method through that a picture will be born-again into its digital kind afterwards performing numerous operations on it. This method is analogous to signal processing, in which the input given is a 2D image, which is a collection of numbers starting from 0 to 255, which denotes the corresponding picture element price. The tactic involves converting a picture into a 2D Matrix.

It consists of 3 common steps[3] :-

1.) **Scanning the image:** a raw image is nonheritable that has got to be processed. It can be expressed in form of pixels as stated above. The aim of this step is to extract info that is appropriate for computing.

2.) **Processing and Enhancing it:** the image is born-again into digital kind by using a digitizer that samples and quantizes the input signals. The speed of sampling ought to be high for good resolution and high quantization level for human perception of different shades using different gray-scale. The obtained result describes the property of the image and additional classifies the image.

3.) **Conversion of Color Image to Gray Scale:** There are basically two methods to convert a color image to a gray scale image :

A.) Average Technique:

In Average technique, the mean is taken of the 3 colors i.e. Red, Blue & Green present in a color image.

$$\text{Grayscale} = (R+G+B)/3;$$

But what happens typically is rather than the grayscale image we tend to get the black image. This is often as a result of we tend to within the born-again image we tend to get 33% every of Red, Blue & Green.

Therefore, to resolve this downside we tend to use the second technique called Weighted technique or Luminosity technique.

B.) Weighted or Luminosity Technique:

To solve the matter in Average Technique, we tend to use Luminosity Technique. In these technique, we decrease the presence of Red Color and increase the color of Green Color and the blue color has the percentage in between these two colors.

Grayscale=((0.3 * R) + (0.59 * G) + (0.11 * B)).

We use this because of the wavelength patterns of these colors. Blue has the smallest wavelength while Red has the utmost wavelength.

4.2 Facial Recognition:

Facial recognition[4] is a process of recognizing a personality's face through technology, it is a system that uses bioscience to map face expression from a photograph or video. It compares the data of the illustrious faces with a database to search out a match. Biometric identification is generally used for security functions.

There are different facial recognition techniques in use, like generalized matching face detection methodology and the accommodative regional mix matching methodology. Most facial recognition systems perform supported the various nodal points on a personality's face. The values measured against the variable related to points of a person's face facilitate in unambiguously distinguishing or supportive the person. With this system, applications will use knowledge captured from faces and may accurately and quickly determine target people. Facial recognition techniques are quickly evolving with new approaches like 3D modeling, serving to beat problems with existing techniques.

There are several benefits related to facial recognition. Compared to alternative biometric techniques, facial recognition is of a non-contact nature. Face picture is captured from a distance and may be analyzed while not ever requiring any interaction with the user/person. As a result, no user will with success imitate another person. Facial recognition will function a wonderful security live for time chase and group action. Facial recognition is additionally low cost technology as there's less process concerned, like in alternative biometric techniques.



Fig.1. Facial Recognition

4.3 Machine Learning:

Machine learning has resolved several issues by selecting one machine learning algorithmic rule, feeding in knowledge and obtaining the result. We have a tendency to don't have to be compelled to build our own neural network. We have access to a trained model that may be used. It will do exactly what we need it to do — outputs a bunch of numbers (face encodings) once we pass within the image of someone's face; comparison face encodings of faces from totally different pictures can tell us if someone's face matches with any image we have.

In straightforward words we can also outline Machine Learning "The ability to learn without being explicit program is named as Machine Learning".

4.4 Python:

Python is a general purpose programming language developed by **Guido van Rossum**, which became very admired in short time because of its simplicity and code readability. It allows the engineer to precise his/her concepts in fewer lines of code while not reducing any readability.

For victorious implementation of this project, the subsequent packages of Python 2.7 have to be downloaded and installed: Python 2.7.x, NumPy, Glob and Random. Python are going to be put within the

default location, C drive during this case. Open Python IDLE, import all the packages and begin operating.

(A). NumPy:

NumPy is one in every of the libraries of Python that is employed for complicated technical analysis. It is used for implementation of multidimensional arrays that consists of assorted mathematical formulas to method. The array declared in a very program encompasses a dimension that is named as axis.

The number of axis present in an array is named as rank

For ex. A= [1,2,3,4,5]

In the given array A five elements are present having rank one, due to one-dimension property.

Let's take another example for higher understanding

i.e. B= [[1,2,3,4], [5,6,7,8]]

In this case the rank is 2 because it is a 2-dimensional array. First dimension has 2 elements and the second dimension has 4 elements.

(B).Glob:

On the basis of the guidelines specified by Unix Shell, the Glob module perceives the pattern and with reference to it, generates a file. It generates full path name.

For.ex.

```
import glob
for name in glob.glob('direc/file*')
print name
```

(C). Random:

Random Module picks or chooses a random range or a component from a given list of parts. This module supports those functions which give access to such operations.

Classification of Random Module: -

- 1) randint(m,n): It returns a value of x such that-
 $m \leq x \leq n$
- 2) randrange(dog, cat, mouse, lion): It returns any stochastic variable or component from the given range .

4.5 OpenCV:

OpenCV is Open Computer Vision Library. I will try to introduce the foremost basic and vital ideas of OpenCV in an intuitive manner. OpenCV is a cross-platform library victimization that we are able to develop real-time computer vision applications. It chiefly focuses on image processing, video capture and analysis as well as options like face detection and object detection.

Using OpenCV[5]:

(A) Read an image: cv2.imread() to read an image. The image must be in the working directory or we can give its full path. We can also specify second argument as flag which specify the way image should be read. cv2.IMREAD_COLOR : It is the default flag ,it loads a color image.

cv2.IMREAD_GRAYSCALE: Loads image in grayscale mode.

cv2.IMREAD_UNCHANGED: Loads image as such including alpha channel.

Ex.

```
import numpy as np
```

```
import cv2
img = cv2.imread('verma.jpg',1)
```

(B) Write an image: `cv2.imwrite()` to save an image, First argument is the file name, second argument is the image you want to save.

Ex. `cv2.imwrite(verma.jpeg,img)`

(C) Display an image:

`cv2.imshow()` to display an image in a window. The window automatically fits to the image size.

Ex. `cv2.imshow(verma,img)`

4.6 Matplotlib:

Matplotlib is a plotting library for Python which provides us a good type of plotting strategies. Here, we are going to find out how to show image with Matplotlib. We are able to zoom pictures, reserve it etc victimisation Matplotlib.

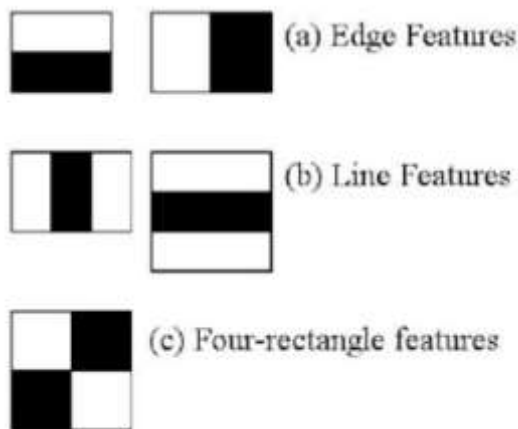
Ex.

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
img = cv2.imread('messi5.jpg',0)
plt.imshow(img, cmap = 'gray', interpolation = 'bicubic')
plt.xticks([], plt.yticks([]) # to cover tick values on X and Y axis
plt.show()
```

4.7 Haar Cascade:

Object Detection victimisation Haar feature-based cascade classifiers is an efficient object detection technique planned by Paul Viola and Michael Jones in their paper, "Rapid Object Detection employing a Boosted Cascade of straightforward Features" in 2001. It is a machine learning primarily based approach wherever a cascade perform is trained from plenty of positive and negative images. It's then accustomed sight objects in alternative pictures.

Here we are going to work with face detection. Initially, the rule needs plenty of positive pictures (images of faces) and negative pictures (images without faces) to coach the classifier. Then we'd like to extract frames from it. For this, haar options shown in below image area unit used. They're rather like our convolutional kernel. Every feature may be a single worth obtained by subtracting add of pixels underneath white parallelogram from add of pixels underneath black parallelogram.



Now all doable sizes and locations of every kernel is employed to calculate lots of features. (Just imagine what quantity computation it needs? Even a 24x24 window results over 160000 features). For every feature calculation, we'd like to search out add of pixels underneath white and black rectangles. To unravel this, they introduced the integral pictures. It simplifies calculation of add of pixels, however giant is also the quantity of pixels, to an operation involving simply 4 pixels. Nice, isn't it? It makes things super-fast.

But among all these options we have a tendency to calculated, most of them are irrelevant. For instance, think about the image below. Prime row shows 2 smart options. The primary feature elect appears to specialize in the property that the region of the eyes is commonly darker than the region of the nose and cheeks. The second feature depends on the property that the eyes area unit darker than the bridge of the nose. However an equivalent windows applying on cheeks or the other place is immaterial. Therefore however can we choose the most effective options out of 160000+ features? It's achieved by **Adaboost**.



For this, we have a tendency to apply each and every feature on all the coaching pictures. For every feature, it finds the most effective threshold which is able to classify the faces to positive and negative. However clearly, there'll be errors or misclassifications. We have a tendency to choose the option with minimum error rate, which implies they're the options that best classifies the face and non-face pictures. (The method isn't as straightforward as this. Every image is given an equal weight within the starting. After each classification, weights of misclassified images are increased. Then again same process is done. New error rates area unit calculated. Additionally new weights. The method is sustained till needed accuracy or error rate is achieved or needed variety of options are unit found).

Final classifier may be weighted add of those weak classifiers. It's referred to as weak because as a result of it alone can't classify the image, however along side others forms a robust classifier. The paper says even 200 options give detection with 95% accuracy. Their final setup had around 6000 options. (Imagine a discount from 160000+ options to 6000 features. That's a giant gain).

So currently you're taking a picture. Take every 24x24 window. Apply 6000 options there to. Check if it's face or not. Wow.. Wow.. Isn't it a touch inefficient and time consuming? affirmative, it is. Authors have a decent resolution for that.

In a picture, most of the image region is non-face region. Therefore it's a more robust plan to possess a straightforward technique to visualize if a window isn't a face region. If it's not, discard it in a very shot. Don't method it once more. Instead specialize in region where there will be a face. This way, we able to realize longer to visualize a doable face region.

For this they introduced the concept of **Cascade of Classifiers**. Rather than applying all the 6000 options on a window, cluster the options into completely different stages of classifiers and apply one-by-one. (Normally initial few stages can contain terribly less variety of features). If a window fails the primary stage, discard it. We have a tendency to don't think about the remaining options there on. If it passes, apply the second stage of options and continue the method. The window that passes all stages may be a face region. However is that the set up !!!

Authors' detector had 6000+ options with 38 stages with 1, 10, 25, 25 and 50 options in initial five stages. (2 options within the higher than the image is truly obtained because the best 2 options from Adaboost). Consistent with authors, on a median, 10 options out of 6000+ are unit evaluated per sub-window.

So this can be a straightforward intuitive clarification of however Viola-Jones face detection works. Scan paper for a lot of details or scrutinize the references in further Resources section.

4.7.1 Haar Cascade victimisation OpenCV:

OpenCV comes with a trainer also a detector. If you would like to coach your own classifier for any object like automobile, planes etc. you'll be able to use OpenCV to form one. Its full details are given here: [Cascade Classifier Coaching\[6\]](#).

Here we are going to influence detection. OpenCV already contains several pre-trained classifiers for face, eyes, smile etc. Those XML files are hold on in `opencv/data/haarcascades/` folder. Let's produce face and eye detector with OpenCV.

First we'd like to load the specified XML classifiers. Then load our input image (or video) in grayscale mode.

```
import numpy as np
import cv2
```

```
face_cascade =
cv2.CascadeClassifier('haarcascade_frontalf
ace_default.xml')
eye_cascade =
cv2.CascadeClassifier('haarcascade_eye.xml'
)
```

```
img = cv2.imread('sachin.jpg')
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
```

Now we discover the faces within the image. If faces are found, it returns the positions of detected faces as `Rect(x,y,w,h)`. Once we have a tendency to get these locations, we can produce a ROI for the face and apply eye detection on this ROI (since eyes are continually on the face !!!).

```
faces=face_cascade.detectMultiScale(gray,
1.3, 5)
for (x,y,w,h) in faces: img =
cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,0)
,2)
```

```
roi_gray = gray[y:y+h, x:x+w]
roi_color = img[y:y+h, x:x+w]
eyes =
eye_cascade.detectMultiScale(roi_gray)
for (ex,ey,ew,eh) in eyes:
```

```
cv2.rectangle(roi_color, (ex,ey), (ex+ew,ey+e
h), (0,255,0),2)
```

```
cv2.imshow('img',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The agenda for this field is to change machines to look at the globe as humans do, understand it during a similar manner and even use the information for a mess of tasks like Image & Video recognition, Image Analysis & Classification, Media Recreation, Recommendation Systems, NLP(Natural Language Processing) etc.

4.8 Convolutional Neural Network:

CNN[9] or ConvNet[10] is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to numerous aspects/objects within the image and be able to differentiate one from the opposite. The pre-processing needed during a CNN is way lower as compared to alternative classification algorithms. whereas in primitive strategies filters are hand-engineered, with enough coaching, CNN have the flexibility to find out these filters/characteristics.

The design of a CNN is analogous there to of the property pattern of Neurons within the Human Brain and was galvanized by the organization of the Visual area. Individual neurons reply to stimuli solely during a restricted region of the sight view referred to as the Receptive Field. A set of such fields overlap to hide the whole cortical region.

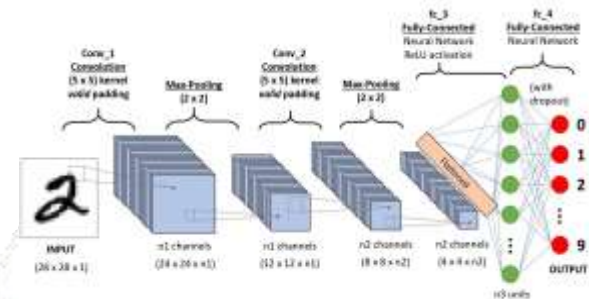


Fig.2. CNN sequence to classify written digits

4.9 TensorFlow:

TensorFlow is associated degree end-to-end open supply platform for machine learning. It's a comprehensive, versatile scheme of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers simply construct and deploy ML powered applications.

TensorFlow is a Python library for quick numerical computing created and discharged by Google.

It is a base library that can be used to create Deep Learning models directly or by using wrapper libraries that alter the method construct on prime of TensorFlow.

4.9.1 How to Install TensorFlow:

Installation of TensorFlow[7],[8] is easy if you have already got a Python SciPy surroundings. TensorFlow works with Python 2.7 and Python 3.3+. You can follow the [Download and Setup directions](#) on the TensorFlow website. Installation is likely simplest via PyPI and specific directions of the pip command to use for your Linux OS or Mac OS X platform are on the Download and Setup webpage.

4.9.2 Keras:

[tf.keras](#) is TensorFlow's high-level API for building and coaching deep learning models. It's used for quick prototyping, progressive analysis and production.

5. DIFFERENT EMOTIONS WHICH WILL BE DETECTED OUT OF ASSOCIATED DEGREE IMAGE:

- Angry
- Surprised

- (c.) Disgusted
- (d.) Sad
- (e.) Happy
- (f.) Neutral



Fig.3.Different

Motions

6. APPLICATION AND FUTURE SCOPE:

Computer Vision is a terribly large field that remains below biological process section. Research work in this field is going at a speedy section.

Emotion detection is an indivisible chunk of computer vision. Countless tasks and processes may be performed if one will become aware of intricacies and endless prospects offered below the sphere of emotion detection.

Some common and wide used applications of emotion detection are:

A. App and product development

Emotion recognition will play an enormous role in optimizing numerous software system engineering processes that includes of testing of ease with that a product may be used. It's a long-established incontrovertible fact that level of comfort with totally different software system merchandise depends massively upon human emotions. A merchandise overall look and feel can even alter human feelings that successively makes the person buy the merchandise or not. Thus, researching concerning totally different emotional states of an individual's body and the way it's influenced by usage of various merchandise could be a matter of prime importance for connected business professionals.

B. Improved learning practices

Proof suggests that a emotional states vouches for higher learning practices whereas the opposite part attempt to suppress them. The distinction between the 2 groups possessing totally different emotional states isn't therefore common to search out. For ex., positive emotion is believed to be dangerous for learning functions whereas slightly negative emotion fosters analytical thinking and is additionally acceptable for polishing off essential tasks.

C. Improvised web development

With the mammoth scale at that the internet is increasing, service suppliers have an interest in assembling tons and tons of knowledge which may be extracted from the users. Correspondingly, all the content and advertisements are played based on the users' profile. Afterward, adding tangled details concerning the various human emotions will offer way more precise behavioral models of various varieties of users.

D. Immersive gaming

Video games represent an oversized chunk of show business. Thus, so as to create these games way more intensive and intrusive, video game creators base their analysis on different kinds of human emotions unremarkably found. So as to attract more and more players, video games are created in such the simplest way that they incorporate human emotions naturally into their game play.

7. CHALLENGES

The main purpose is to discover numerous emotions in a given sample image. The foremost difficult half during this task was to see the precise feeling once 2 emotions look quite similar, for e.g. "Disgust" being classified as "Sadness", "Surprise" like "Happy" then on. Currently for 8 totally different classes, the result was roughly 70% accurate that is kind of well really as our classifier learned quite an bit. So, we have a tendency to should see however will we have a tendency to increase its potency and accuracy.

If we glance at our emotions list we will decide that we've got solely restricted variety of examples for "sad", "fear" and "contempt". By increasing the amount of pictures for these emotions we will definitely increase improvement, or if we have a tendency to not contemplate these emotions within the list then improvement may be exaggerated over 80%.

8. CONCLUSION

If the user is feeling sad, then system will mechanically display some message along with user's mood on the user's screen. System is able to send some links, web pages or information by recognizing response from user. It will help to decrease level of stress on mind. Also support stress management.

9. ACKNOWLEDGEMENT

I feel great pleasure in acknowledging the help given by various individuals throughout the project work. This project is itself an acknowledgement to the inspiration, drive and technical assistance contributed by many individuals.

I take this opportunity to express my immense gratitude to my project guide Mr. Prabhat Chandra Gupta in my project "DESKTOP PARTNER WITH PYTHON & ML". I am grateful for their prolonged interest in my work and excellent guidance. They have been a constant source of motivation to me.

10. REFERENCES

- [1] B. Fasel and J. Luetttin. Automatic facial expression analysis: a survey. *Pattern recognition*, 36(1):259–275, 2003.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [3] <http://bvicam.ac.in/news/INDIACom%202018%20Proceedings/Main/papers/3248.pdf>
- [4] <https://medium.com/datadriveninvestor/machine-learning-on-facial-recognition-b3dfba5625a7>
- [5] <https://www.pyimagesearch.com/2018/05/28/ubuntu-18-04-how-to-install-opencv/>
- [6] https://www.bogotobogo.com/python/OpenCV_Python/python_opencv3_Image_Object_Detection_Face_Detection_Haar_Cascade_Classifiers.php

- [7] <https://www.tensorflow.org/install/pip>
- [8] <https://www.tensorflow.org/install>
- [9] <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>

- [10] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images, 2009.

