Emotion detection system/Pypower project

3 steps in this project:

Collecting dataset for various emotion:

Created 5 classes for 5 different emotions:

Angry happy Neutral Sad Surprise

Applying deep learning for training the model:

Import the required libraries

Use the images and labels for training the model

Train the model using keras and tensorflow based on given parameters

Using the model to predict the emotion:

Faceclassifier file(xml file) to detect the face

Emotion\_detection.h5 file(pertained model) to detect the emotion(it is keras trained model)

H5 extension is used for trained model files

Test.py(python file) used to execute the project

In this project there are total 5 emotions:

Angry happy Neutral Sad Surprise

2nd project:

Emotion detection system

Matlab

5 emotions classes:

Anxiety disgust happiness and fair

60% to 70% accuracy achieved till now.

3rd project:

Emotion detection system

Pretrained kaggle model for emotions

Author of pretrained model jonathan oheix.

Multiple detection accuracy decreases with increase in more number of faces

70% accuracy achieved till now.

For using matplotlib, images from trained model are need to be sequentially saved.

Images are usually formed in grayscale format.

References

<https://www.youtube.com/watch?v=PulKlAZRoAY>

<https://www.youtube.com/watch?v=bYvNY8TqIO4>

<https://www.youtube.com/watch?v=Bb4Wvl57LIk>

<https://www.youtube.com/watch?v=fkgpvkqcoJc&t=80s>

<https://www.youtube.com/watch?v=avv9GQ3b6Qg>

<https://www.youtube.com/watch?v=0O0otPBtbXs>

<https://www.youtube.com/watch?v=ou-uhY4616A>

libraries:

matplotlib

numpy

pandas

cv2

keras(deep learning libraries)

seaborn

keras.layers(to import layers as in artificialneural networks)

keras.

**Active contour** is a type of **segmentation** technique which can be defined as use of energy forces and constraints for segregation of the pixels of interest from the **image** for further **processing** and analysis. **Active contour** described as **active** model for the **process** of **segmentation**.

**BLOB** stands for Binary Large OBject and refers to a group of connected pixels in a binary **image**. The term “Large” indicates that only objects of a certain size are of interest and that “small” binary objects are usually noise.

**Edge** detection is an **image processing** technique for finding the boundaries of objects within **images**. It works by detecting discontinuities in brightness. **Edge** detection is used for **image segmentation** and data extraction in areas such as **image processing**, computer vision, and machine vision.

In **image processing**, a **Gabor filter**, named after Dennis **Gabor**, is a linear **filter** used for texture analysis, which essentially means that it analyzes whether there is any specific frequency content in the **image** in specific directions in a localized region around the point or region of analysis.

**PCA** is a pre-**processing transformation** technique that creates new **images** from the uncorrelated values of different **images** [13]. ... **PCA** is used to find- out principal components in accordance with maximum variance of a data matrix.

Grayscale is a range of [monochromatic](https://techterms.com/definition/monochrome) shades from black to white. Therefore, a grayscale image contains only shades of gray and no color.

While [digital](https://techterms.com/definition/digital) images can be saved as grayscale (or black and white) images, even color images contain grayscale information. This is because each [pixel](https://techterms.com/definition/pixel) has a luminance value, regardless of its color. Luminance can also be described as brightness or intensity, which can be measured on a scale from black (zero intensity) to white (full intensity). Most image [file formats](https://techterms.com/definition/file_format) support a minimum of 8-bit grayscale, which provides 2^8 or 256 levels of luminance per pixel. Some formats support 16-bit grayscale, which provides 2^16 or 65,536 levels of luminance.