Project Description
Face Emotion Recognition

Vikram S. Bajaj

7th September, 2016

Table of Contents

First Try

Dataset Description

The dataset used in the first trial was the CMU Face Images Data Set from the UCI Machine Learning Repository.

Link: https://archive.ics.uci.edu/ml/datasets/CMU+Face+Images

The dataset originally contained 20 directories, one for each person, named by userid. Each of these directories contains several different face images of the same person.

Naming Convention Used

The images in each directory were named as:

userid_headposition_emotion_eyestate_scale.pgm

userid	:	an2i, at33, boland, bpm, ch4f, cheyer, choon, danieln, glickman, karyadi, kawamura, kk49, megak, mitchell, night, phoebe, saavik, steffi, sz24, and tammo
headposition	:	straight, left, right, up
emotion	:	neutral, happy, sad, angry
eyestate	:	open, sunglasses
scale	:	1 (full resolution: 128x120), 2 (64x60), and 4 (32x30)

Changes Made

Instead of having 20 directories named by userid, I re-categorized the images based on emotion. So, I grouped the images into 4 directories: angry, happy. neutral, sad.

Reason For Not Using This Dataset

SURF+BOW features were extracted and SVM with Gaussian Kernel was used for classification. Although it achieved 98% accuracy on training set, it achieved only 11% accuracy on the validation set. Also, it contained only 4 emotions.

Second Try

Dataset Description

The dataset used was the JAFFE dataset.

It contained 213 images belonging to 7 categories of expressions - anger, disgust, fear, happiness, sadness, surprise, neutral.

I categorized the images into the 7 categories based on their naming convention.

Using all 7 categories for training, SURF+BOW features were extracted and an SVM with Gaussian Kernel was used for classification.

It achieved 100% accuracy on training set and 59% accuracy on validation set. The worst misclassification was that of sadness.

Using only 4 categories - anger, happiness, sadness, neutral - and using Linear SVM instead of Gaussian kernel SVM, training set accuracy was 100% and validation set accuracy was 83%. (.mat files have been uploaded for this case).

Issue With This Dataset

Didn't seem to generalize well to test images, especially those depicting sadness.

Third Try

Dataset Description

The KDEF Dataset was used.

Expressions: 7 (neutral, happy, angry, afraid, disgusted, sad, surprised).
Angles: 5 (-90, -45, 0, +45, +90 degrees: i.e. full left profile, half left profile, straight, half right profile, full right profile).
Number of pictures: 4900 approx
Size: 562 * 762 pixels.
Codes:
File Name Example: AF01ANFL.JPG
Letter 1: Session
A = series one
B = series two

Letter 2: Gender
F = female
M = male

Letter 3 & 4: Identity number
01 - 35

Letter 5 & 6: Expression
AF = afraid
AN = angry
DI = disgusted
HA = happy
NE = neutral
SA = sad
SU = surprised

Letter 7 & 8: Angle
FL = full left profile

HL = half left profile
S = straight
HR = half right profile
FR = full right profile

I sorted the images into 7 folders based on emotion categories, depending on 5th & 6th letters of file name.

Problems While Learning

Due to the large number of images (min 554) in each category, MATLAB ran out of memory while running surf_bow.m.

Error: *Error using zeros*

Requested 57796480x64 (13.8GB) array exceeds maximum array size preference. Creation of arrays greater than this limit may take a long time and cause MATLAB to become unresponsive. See array size limit or preference panel for more information.

Solution Try 1

I tried using only 4 emotion categories: happy, sad, angry, neutral.

Error: *Error using zeros*

Requested 33281920x64 (7.9GB) array exceeds maximum array size preference. Creation of arrays greater than this limit may take a long time and cause MATLAB to become unresponsive. See array size limit or preference panel for more information.

Solution Try 2

I tried using 40% of images for training (this came to 223 images per category) instead of 70%, using 4 emotion categories (happy, sad, angry, neutral).

This didn't give any errors while training. However, the classification accuracies on training sets and validation sets were 90% and 51% respectively (using SVM with Gaussian Kernel).

The model also didn't perform well on test examples, and seemed to be biased towards angry.

Reason For Not Using This Dataset

On further investigation, I noticed that majority of the categories had no front facing images as samples. This could probably be the reason behind its poor performance on test examples, which had faces facing towards the camera.

Fourth Try

Dataset Description

The Kaggle Dataset of Face Expressions (From the Challenges in Representation Learning: Facial Expression Recognition Challenge) was used.

The data consists emotion labels and pixel information of 35,887 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centered and occupies about the same amount of space in each image. Seven categories: 0=Angry, 1=Disgusted, 2=Afraid, 3=Happy, 4=Sad, 5=Surprised, 6=Neutral.

fer2013.csv contains two columns, "emotion" and "pixels".

The "emotion" column contains a numeric code ranging from 0 to 6, inclusive, for the emotion that is present in the image.

The "pixels" column contains a string surrounded in quotes for each image.

To generate images from the pixel information, I did the following:

The dataset had rows with emotion (0-6) in random order. So, I sorted the rows based on the emotion label. This way, records of a class were grouped together. This was then stored in a file called sorted_data.mat.

From this file, I generated angry.mat, disgusted.mat, afraid.mat, happy.mat, sad.mat, surprised.mat and neutral.mat and then used generate_images.m to generate images from these mat files.

These files are available upon request.

Fifth Try

Dataset Description

I created my own dataset (with 7 emotions, including neutral) using Google Image Search results.

All the images have dimensions of 150x150 pixels. This dataset is available upon request.

However, I wasn't able to achieve a very high accuracy using this dataset.