

Branch: master ▾

[MLND](#) / [projects](#) / [facial_recognition_family_members](#) / [proposal.md](#)

Find file

Copy path

 **lxwvictor** Update proposal.md

b97decc a minute ago

1 contributor

51 lines (31 sloc) 4.52 KB

Machine Learning Engineer Nanodegree

Capstone Proposal

Li Xiaowei 24/12/2017

Proposal

Domain Background

In the recent years, many consumer grade products have started to deploy face recognition features. One of them is to set the name of faces in my iPhone iCloud photo library. Many beautiful memories about a specific person will be created. For example, how my daughter grew up from her born to now's four years old. When I want to view a photo about my daughter, my phone is able to show me all related photos.

I have been taking photos since my college life. It has been more than 10 years and I have accumulated more than 40k photos. I always wanted to do something to auto preprocess my photos when I import them into my computer. In this project I want to explore how to use machine learning or deep learning to perform facial recognition. More specifically, I want the faces of my family members can be tagged when a photo is imported in.

Problem Statement

Given the photo library I have, to extract/detect the faces first. The cropped/extracted face images will be learnt together with the labels (my family members) and then auto tag the name of the faces for unseen photos. This project is not targeted for large or internet scale data.

Datasets and Inputs

Despite I have a large photo library, not all of them are related to this project. For example, photos with no human face, photos with people other than my family members.

OpenCV will be employed to extract the faces and save them as new files to be used for training later. The original files are at least 10M pixels, there is enough room for cropping and resizing face images to 299 * 299 to fulfill the input requirement later. In addition, there should be at least near a hundred images for each label. Depending on the performance, techniques of using image generator to produce more input images may be needed.

The input will be cropped face images, instead of the entire image which is too distracted for the purpose of face recognition.

Solution Statement

The first step in this project will be using openCV or other possible approaches to extract the faces from my photo libraries. My photo library is in chronological order and with proper naming about where, who and what. So it won't be too difficult to hand pick all the related faces. There will be seven people/classes for the dataset.

Performing training on images is always good to start with extracting "features". So the second step will be using Google Inception V3 to extract the features. Each face photo will be represented by a vector of 4096 elements.

The dataset will be splitted into training and testing datasets at the third step.

At the fourth step, I'll explore to use KNN alone, PCA with KNN to evaluate the performance of accuracy and speed. DNN will also be explored, and basically it's a transfer learning here.

The fifth step is the summary of the performance about the recognition accuracy and speed.

Benchmark Model

At least three different models will be explored, named KNN, PCA + KNN and DNN transfer learning. The baseline of the performance will be based on KNN. Both accuracy and speed will be evaluated in this report.

Evaluation Metrics

The metrics will focus on accuracy and speed. Accuracy will be based on the confusion matrix of predicted and real classes. Speed will be evaluated by the system time taken by predication.

It's quite straight-forward to choose accuracy as one of the metrics as the objective is to have a model able to make as many correct predictions as possible. The speed doesn't matter much for this project as the scale is quite small. But it's an unavoidable factor if considering to re-scale to industrial/commercial use.

Project Design

Unlike other projects in this MLND, which were given the input dataset. This project may require great amount of time to "create" the input dataset first.

So the entire project will have three main components.

The first one is to extract faces from my photo library and manually pick/group the face photos to prepare the input dataset. There will be seven classes.

And The second component will cover the training and performance of different models including KNN, PCA + KNN and DNN with transfer learning. The performance of KNN will be set as the baseline. PCA + KNN is more likely to outperform on the speed. But I guess DNN with transfer learning should find the best balance of accuracy and speed.

The final component will focus on the analysis and comparison of performance from different models.