MACHINE LEARNING ENGINEER NANODEGREE

José Piñero
August 3rd, 2017

Proposal

Domain Background

This project is going to be enclosed in the Computer Vision domain. Computer Vision is an interdisciplinary field that seeks to automate tasks that the human visual system do, vision tasks such as gaining full understanding of a scene or environment. In order to accomplish such visual tasks, feature extraction and processing is needed in order to ultimately gain high-dimensional understanding from the presented data.

Precisely for this project I will be working on face recognition, a task where humans tend to have 97% or more precision, but it hasn't been that easy for machines until recently. A lot of work have been put into this task and many algorithms have been developed to address it from classical neural networks [1], genetic algorithms [2], principal component analysis [3] among others. Recently a new approach has arisen named Convolutional Neural Networks, this algorithm have shown exceptional precision in vision tasks outperforming the traditional techniques.

Problem Statement

The problem that I'm going to address in this project is the face recognition by using Convolutional Neural Networks (CNN). Given a picture of a face, correctly identify it. This is a very challenging task due to the similarity between some persons. The general features are the same: nose, eyes, ears, mouth, etc, and that's why the algorithm must learn even more detailed features in order to correctly classify the person in it. I'm going to pass a set of pictures containing faces through the algorithm, in order for it to learn them and then try to correctly classify the known persons in any other pictures of their faces. This is a multi-class classification problem, and thus the performance of the algorithm can be measured using well known metrics such as precision or cross entropy, defined below:

$$Cross - Entropy = -\sum_{i=1}^{n}\sum_{j=1}^{m} y_{i,j} * \lg(p_{i,j})$$

Where:
- *n* stands for the number of instances in the dataset
- *m* stands for the number of different identities in the dataset
- $y_{i,j}$ is the true probability of the instance *i* to have the identity *j*
- $p_{i,j}$ is the predicted probability of the instance *i* to have the identity *j*

Datasets and Inputs

The dataset consist of around 8000 pictures containing faces from celebrities. For each celebrity there are around 100 pictures, the faces may be in different angles, have different hair styles, skin tones and may wear hat or glasses. Here's an example:

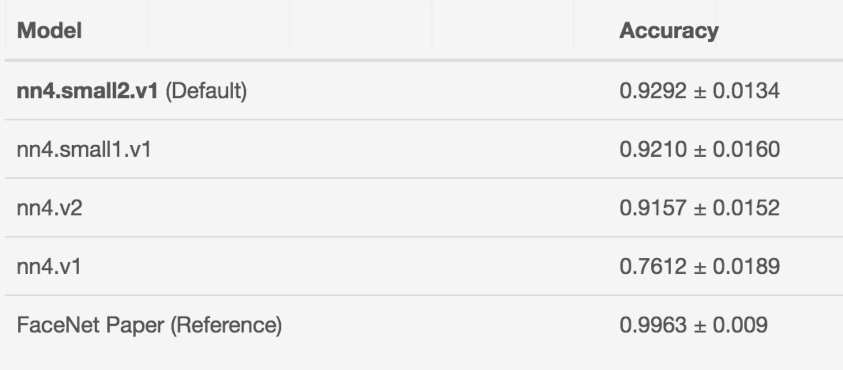
The dataset have been created manually by searching public pictures from 80 celebrities and then automatically cropping them by using the Viola-Jones algorithm to keep only the face. The 100 obtained pictures from each celebrity is going to be split into training and testing set. The CNN is going to train on the first set and then it's going to try to correctly classify the celebrities from the testing set. The dataset has been intentionally created for the task.

Solution Statement

The intended approach to tackle this problem is to use Convolutional Neural Networks (CNN). This technique has been proved to be powerful computing visual tasks, but CNN's require a lot of time and computation power in order to achieve their best performance, for that reason the idea wouldn't be to create a CNN from scratch but try to transfer the weights and features, already learned from famous CNN's architectures, to this problem by doing some light retrain and fine tuning parameters. The input to the algorithm are going to be the images (probably scaled in order to fit the input constraints 244x244) and the output from the net is going to be a vector with 80 floating point values (1 per class) where each value represents the predicted probability of the picture to belong to the *i-th* identity.

Benchmark Model

One of the best models that can be found is the FaceNet model developed by Google [4]. And it's accuracy can be found in the following table alongside some FaceNet variants:



Model	Accuracy
nn4.small2.v1 (Default)	0.9292 ± 0.0134
nn4.small1.v1	0.9210 ± 0.0160
nn4.v2	0.9157 ± 0.0152
nn4.v1	0.7612 ± 0.0189
FaceNet Paper (Reference)	0.9963 ± 0.009

It can be noticed that the models have obtained remarkable accuracy. The FaceNet algorithm have outperformed even humans. All of this results are reported on the public available data LFW (Labeled Faces in the Wild). Probably I won't be able to obtain such high accuracy however I expect to get 60% or more by using transfer learning on my dataset. In order to compare the results

Evaluation Metric

The performance can be ultimately measured using the precision, defined as below:

$$precision = \frac{1}{n}\sum_{i=1}^{n} V(i) \qquad \text{where } V(i) = \begin{cases} 1 & if y_i = arg\,max(p)) \\ 0 & otherwise \end{cases}$$

- y_i is the true identity of the instance *i*
- p is the predicted vector from the CNN

Project Design

The project consists on using transfer learning from a pre-trained CNN to make it work on my own dataset. The project will have the following steps:

- Perform data augmentation by rotating the images and applying other transformations in order to increase the size of the training data and help the algorithm to capture invariants.
- Choose a pre-trained model and test its performance on the dataset.
- Improve the performance of the model by perhaps retraining or fine tuning some parameters, optimizers, regularizers, etc.
- Compare the results on the dataset with the best models available (FaceNet or its variants) and report the metrics and results obtained.

References

[1] T. Kohonen. Self-organization and Associative Memory. Springer-Verlag, Berlin, 1989.

[2] P. J. B. Hancock and L. S Smith. *GANNET: Genetic design of a neural net for face recognition. In H-P. Schwefel and R. Manner*, editors, Proceedings of the Conference on Parallel Problem Solving from Nature. Springer Verlag, 1991.

[3] L. Sirovich and M. Kirby. *A Low-Dimensional Procedure for the Characterization of Human Faces*, J. Optical Soc. Am. A, 1987, Vol. 4, No.3, 519-524.

[4] Florian Schroff, Dmitry Kalenichenko and James Philbin. *FaceNet: A Unified Embedding for Face Recognition and Clustering*. 2015